Keytool and Certificate Management

A guide to utilizing keytool to assist with Certificates for eMedNY SOAP

STATE OF NEW YORK
DEPARTMENT OF HEALTH

6/1/2013

TABLE OF CONTENTS

1 Introduction 3
2 Creating a Certificate Signing Request (CSR) and requesting a Certificate 4
2.1 Create the "KeyStores" Folder & Open the Command Window 4
2.2 Generate a key pair for your web service client 6
2.3 Generate a certificate request 7
2.4 Submit the Certificate Signing Request through ePACES 7
2.5 Retrieve the Client Certificate from ePACES 8
3 Importing the Certificate 10
3.1 Importing the Client Certificate into the keystore 10
3.2 Importing the Server Certificate 11
3.3 Importing certificates using a MMC 12
3.4 Importing Certifcates Using the IE Certificates Wizard 19
3.5 Importing the Server Certificate into the Keystore 27
3.6 Importing private key and certificates from Java to Windows Key Stores 29
4 Additional Tools and Information 34
4.1 keytool web link 34
4.2 Requirements for CORE Compliance 34
4.3 JSSE Reference Guide 34
4.4 WCF – 2 Way SSL using Certificates 34

For eMedNY application support and general application questions, please contact the eMedNY End-User Help Desk at 1-800-343-9000 or eMedNYAppHelpDesk@csc.com.

1 Introduction

This document is intended to assist providers with the acquisition and use of security certificates when accessing secured eMedNY Web Services (such as the File Transfer Service and the Meds History Service.) It details the processes needed to create a private and public key pair, and an associated Certificate Signing Request (CSR) and how to process the resulting signed certificate delivered in response from eMedNY, as well as how to import any server certificates that may also be required (e.g MedsHistory Web Service).

The process to enroll as a user of eMedNY-signed certificates , to submit the Certificate Signing Request created as described in this document, and the retrieval of the user's eMedNY-signed certificate are detailed in a separate document eMedNY X509 Certificates Guide.

Please note that the information provided is to be used as a guide only. It will be of greatest interest to developers using Java or Dotnet technologies on Windows platforms. Developers on Linux-variants should be able to adapt the instructions given herein. Developers using key and certificate stores other than those covered in this document will have to adapt the information here to the fit their own situation.

2 Creating a Certificate Signing Request (CSR) and requesting a Certificate

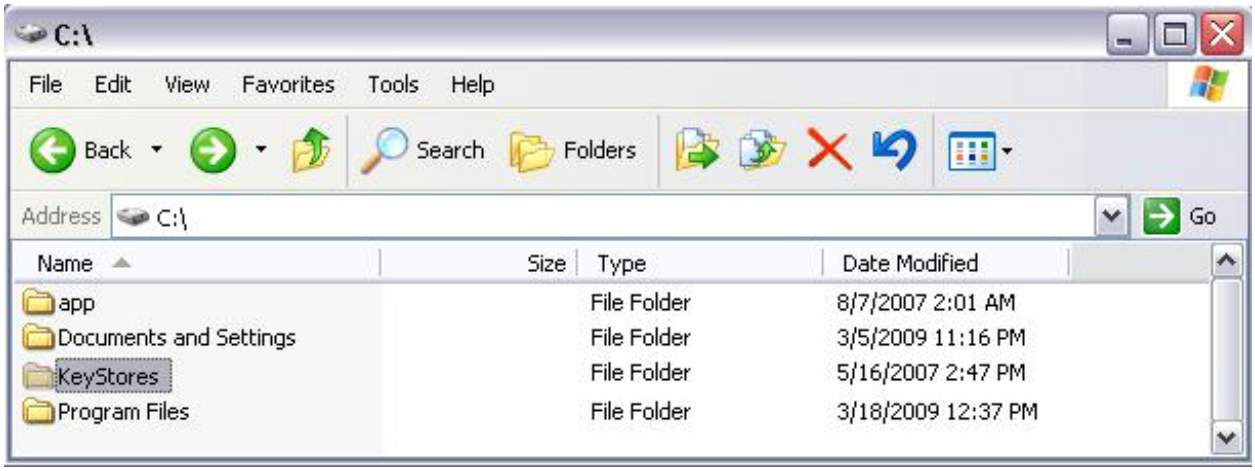
"Keytool" is a key and certificate management utility, which will be used to generate key pairs for your web services client. This utility will need to be installed so that it can be run in any directory. For additional information regarding keytool, see keytool web link in Section 4.1. The following instructions are for Windows-based operating systems. Other operating systems have analogous tools; users should consult the appropriate references or with your IT departments.

eMedNY can provide scripts contained in batch files with these commands. They can be run using instructions contained within these files. Please email emednyproviderservices@csc.com to request these scripts.

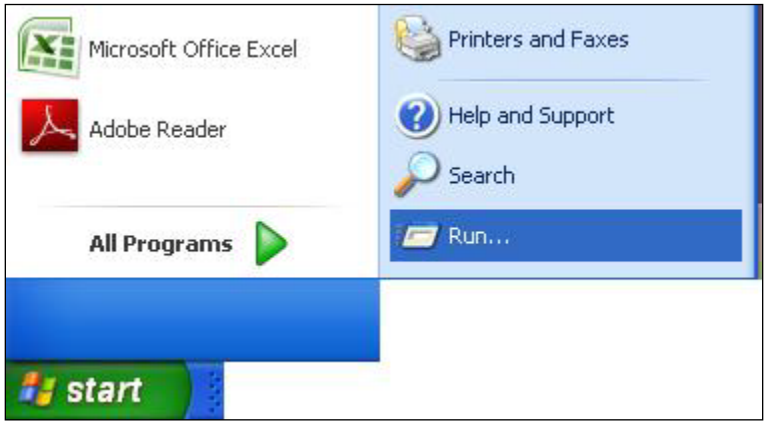
The following instructions are for example purposes only. Words that are in bold are sample text, please use whatever suits the needs of your software in its place.

2.1 Create the "KeyStores" Folder & Open the Command Window

In Windows Explorer you will need to create a new folder for your keytool-related files. In our example, we have called it **KeyStores**:



Next, In Start Menu Click Run:



Type “cmd” and press Enter:

Change directory to the one you created earlier (i.e. **Keystores**):

2.2 Generate a key pair for your web service client

At the command line, type in the following command, substituting your own data for the text in brackets []. A full explanation of this data appears after the command line:

keytool -genkeypair -v -alias [Client Alias] -keystore [keystore name].jks -keyalg RSA -sigalg SHA1withRSA -storepass [keystore password] -dname "[Client's DN, see below]" -keypass [client password]

Client Alias – An alias set up by the user. It is used to refer to the keystore. A keystore can contain multiple client items, each referenced by its own unique alias.

Keystore Name – The name of the Keystore. A file named [keystore name].jks will be generated.

Keystore Password – A password to access the keystore. We suggest a password should be at least six characters in length and contain at least one number and punctuation mark.

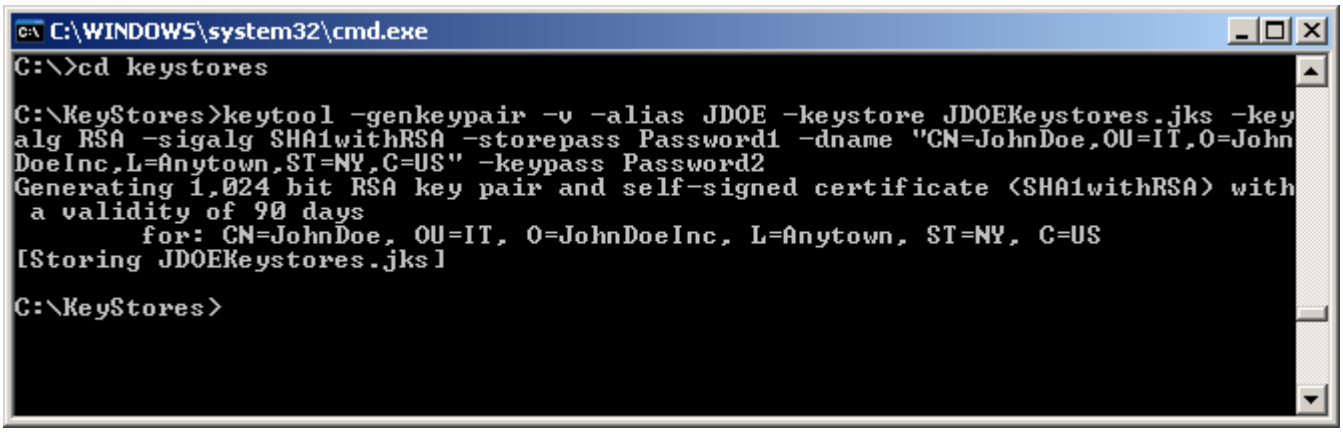
Client's DN – This is information identifying the client. You should populate it with information identifying your organization (using no spaces) as follows: "CN = [Client Name], OU=[Organizational Unit], O=[Organization], L=[City or Locality], S=[State], C=US"

Client Password – The client password to access the private key in this keystore.

This is an example:

keytool -genkeypair -v -alias **JDOE** -keystore **JDOEKeystores**.jks -keyalg RSA -sigalg SHA1withRSA -storepass **Password1** -dname "CN=**JohnDoe**,OU=**IT**,O=**JohnDoeInc**,L=**Anytown**,ST=**NY**,C=**US**" -keypass **Password2**

See below for an example:


C:\WINDOWS\system32\cmd.exe
C:\>cd keystores
C:\KeyStores>keytool -genkeypair -v -alias JDOE -keystore JDOEKeystores.jks -keyalg RSA -sigalg SHA1withRSA -storepass Password1 -dname "CN=JohnDoe,OU=IT,O=JohnDoeInc,L=Anytown,ST=NY,C=US" -keypass Password2
Generating 1,024 bit RSA key pair and self-signed certificate (SHA1withRSA) with a validity of 90 days
for: CN=JohnDoe, OU=IT, O=JohnDoeInc, L=Anytown, ST=NY, C=US
[Storing JDOEKeystores.jks]
C:\KeyStores>

2.3 Generate a certificate request

Type the following command to generate a Certificate Signing Request. Replace the text in brackets with the same data you used to generate the Keystore.

keytool -certreq -alias [Client Alias] -keystore [Keystore Name].jks -file [Client Alias].csr -storepass [Keystore Password] -keypass [Client Password]

See below for an example:

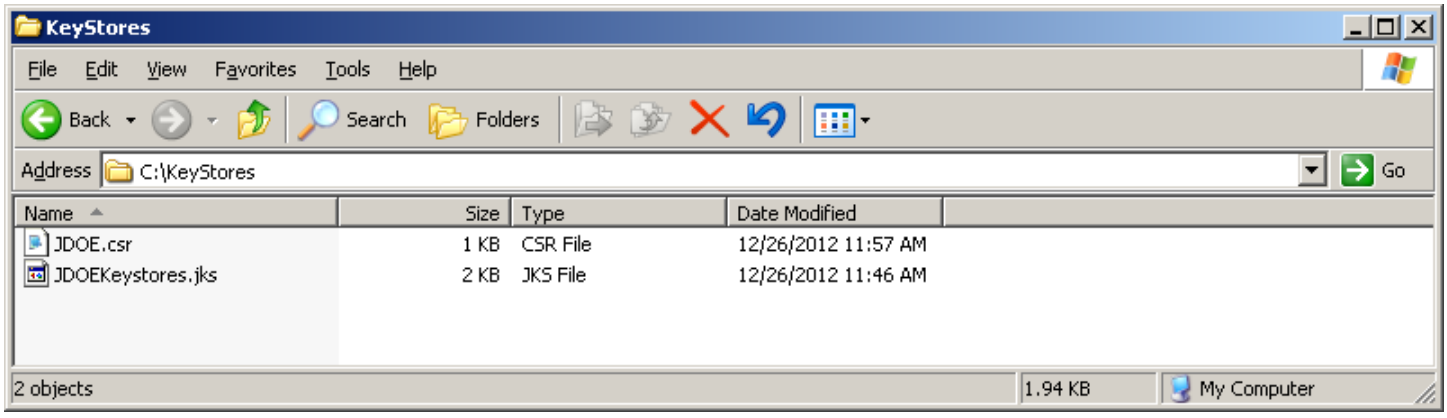
keytool -certreq -alias **JDOE** -keystore **JDOEKeystores**.jks -file **JDOE**.csr -storepass **Password1** -keypass **Password2**

The image follows that example:

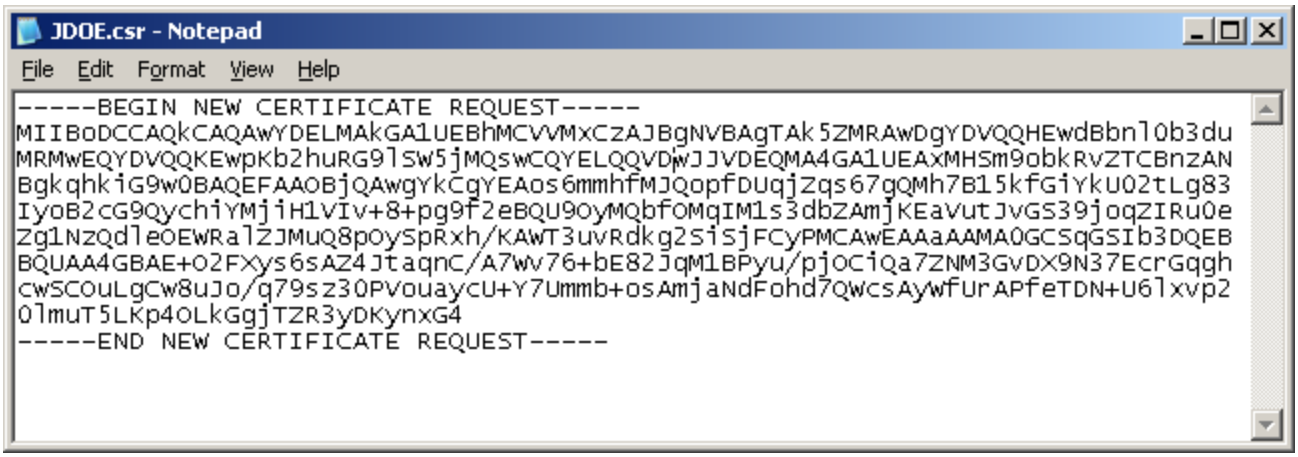
This will result in a Certificate Request file (.csr) being placed into the same directory.

2.4 Submit the Certificate Signing Request through ePACES

View the **Keystores** directory in Windows explorer.



As you can see above, there are two files in the directory. The .jks is your keystore file. The .csr is your Certificate Signing Request. Open the .csr file in a text editor, such as NotePad. It should look something like this:

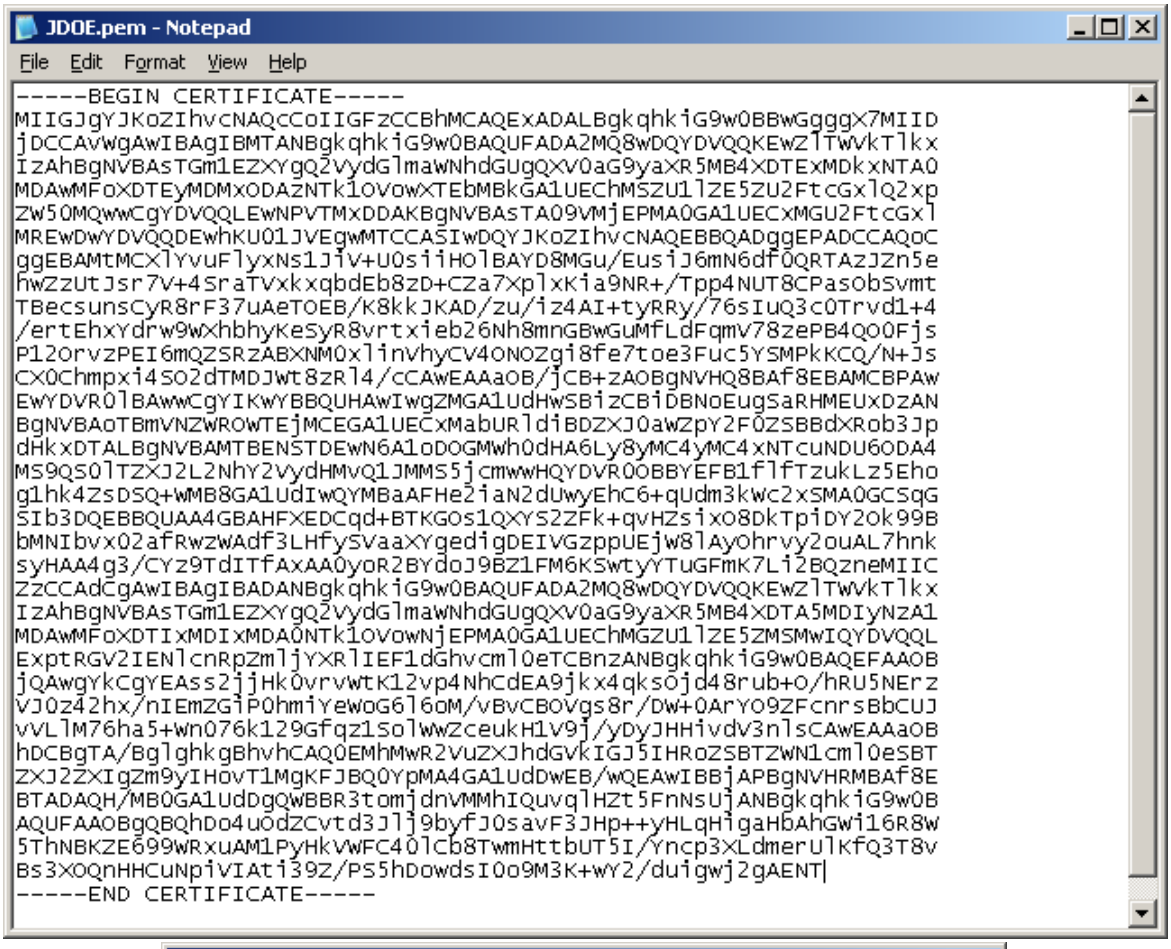


Copy this information into the "Certificate Signing Request" Field in ePACES as per the instructions in the eMedNY X509 Certificates Guide.

2.5 Retrieve the Client Certificate from ePACES

Retrieve the Client Certificate through ePACES as per the instructions in the eMedNY X509 Certificates Guide. Copy the data there into Notepad and save it in the "Keystores" directory under the name [Client Alias].pem

This is a sample of a Client certificate.



3 Importing the Certificate

3.1 Importing the Client Certificate into the keystore

Type the following command to import the Client Certificate into your keystore. Replace the text in brackets with the appropriate data – [Cert file name] is the name of the file where you saved the certificate from ePACES. An explanation of the data elements follows the command:

keytool -importcert -v -alias [Client Alias] -file [Cert file Name].pem -keystore [Keystore name].jks -storepass [keystore password] -keypass [client password]

Client Alias – The same Client Alias as above.

Cert File Name – the Name of the desired certificate file. We recommend the same as the Client Alias for tracking purposes.

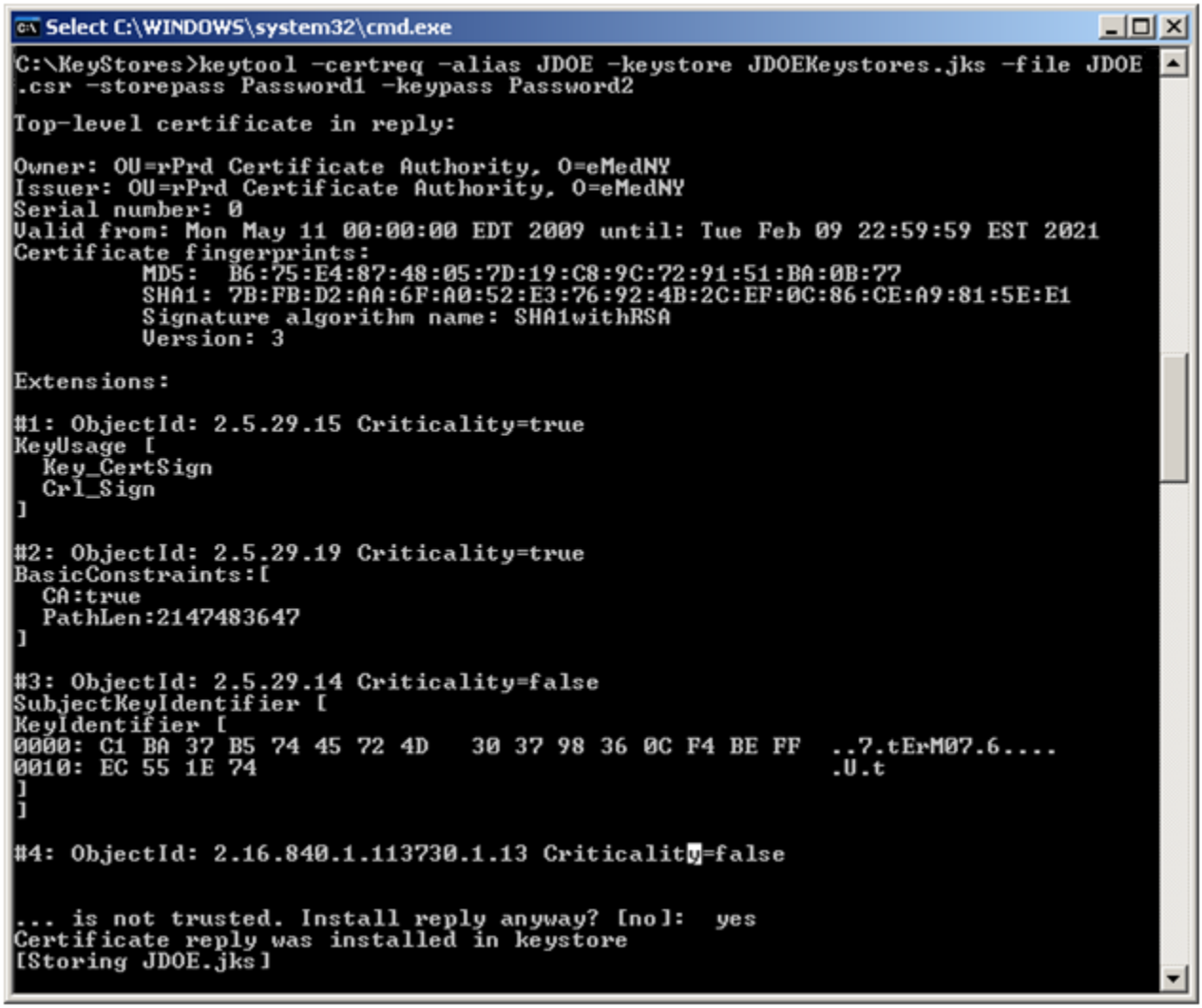
Keystore Password – A password to access the keystore. We suggest a password should be at least six characters in length and contain at least one number and punctuation mark. This is the same as used for creating the keystore and generating the CSR.

Client Password – The client password to access the private key in this keystore.

This is an example command:

keytool -importcert -v -alias **JDOE** -file **JDOE**.pem -keystore **JDOEKeystores**.jks -storepass **Password1** -keypass **Password2**

The following image follows our example:



The reason the certificate is indicated as “not trusted” as the certificate returned by eMedNY is a chain consisting of the user’s cert as well as the Certificate Administrator Signer Cert (rPrd Certificate Authority.) It is the latter certificate that is “not trusted.” You can also import only the eMedNY client cert without the CA signer cert if you first import the returned chain into Internet Explorer and then re-export just the eMedNY client cert from the browser. If you need to do this for your software, see the instructions under Importing Certificates using the IE Certificates Wizard.

3.2 Importing the Server Certificate

For Java based web service clients, doing message level security (e.g. Meds History Service - MHS,) the Server certificate needs to be imported into this keystore or in a separate keystore. We will do it in this keystore.

The Server certificate can be obtained from eMedNY – please email emednyproviderservices@csc.com if you need this certificate. For this demo, it will be downloaded to:

C:\Keystores\dev_server.pem

And imported using KeyToolIUI via Import > Keystore's entry > Trusted Certificate > Regular Certificate

However, this will fail. This is because the eMedNY cert contains a certificate chain (server cert + CA cert) and a jks keystore will not allow one to import a cert as a trusted cert if it contains a chain. We will need the server certificate without a chain (i.e remove the CA cert.) Note we could import the client cert **jsmith.cer** because the keystore contains its private key and we imported into an existing entry and replaced the self-signed cert in it with the **jsmith** cert + CA cert. However as we do not have the server's private key we cannot do the same. (Other alternatives are also possible such as creating a pkcs12 keystore, or obtain the signer certficate and importing it as a trusted cert. However the process given in this document will be applicable to both Java and Dotnet developers on Windows platforms.)

The way around this is to first import the eMedNY Server certificates (servercert + cacert) into the Windows Certificate Store. Then export only the servercert from the Windows Certificate Store and then import the servercert into the keystore as a trusted cert.

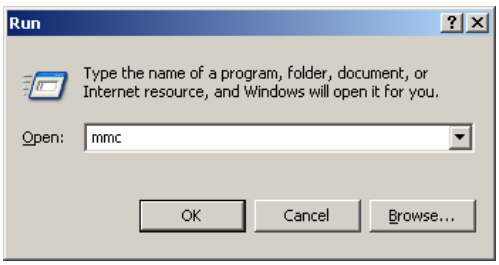
There are two ways of doing this, the first giving a more detailed view into the Windows Certificate Store, while the second is less granular but simpler to use.

- Using a Microsoft Mangement Console (mmc) for the Windows Certificate Stores
- Using the Internet Explorer Certificates Wizard.

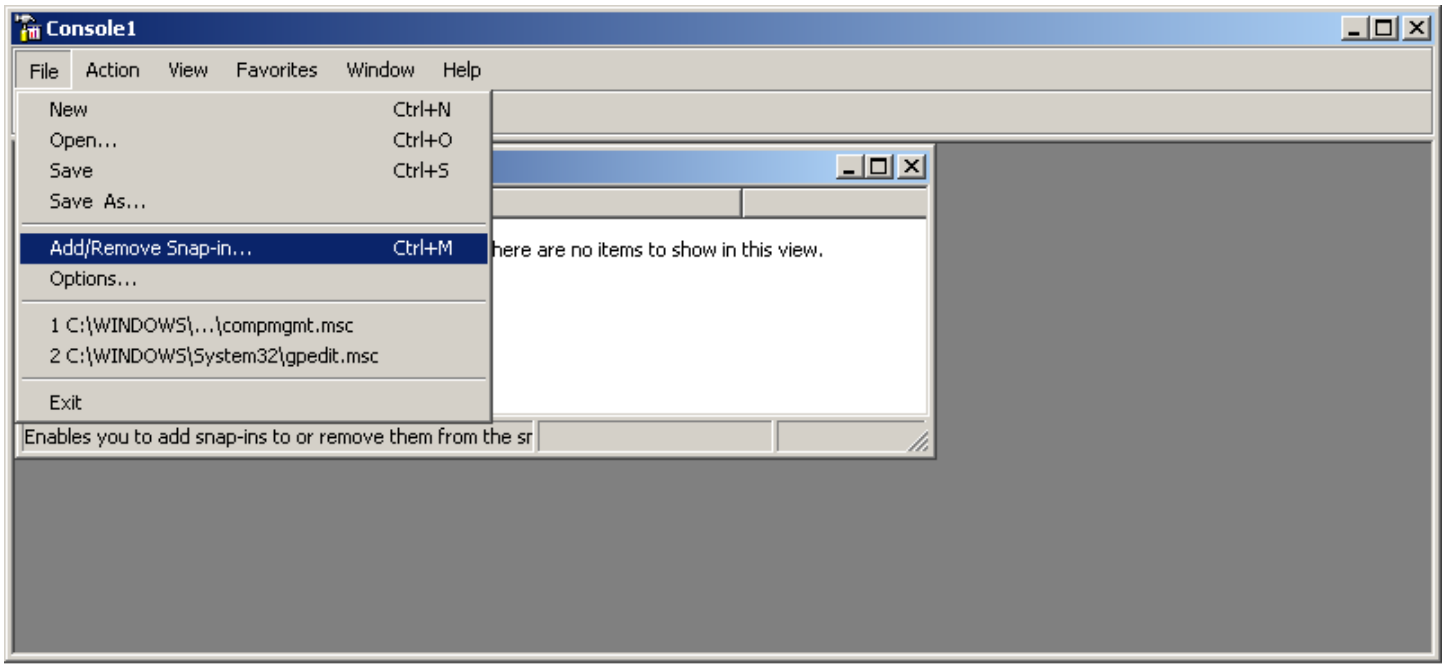
3.3 Importing certificates using a MMC.

First create a Microsoft Management Console (mmc) for the Certificate Store by doing the following:

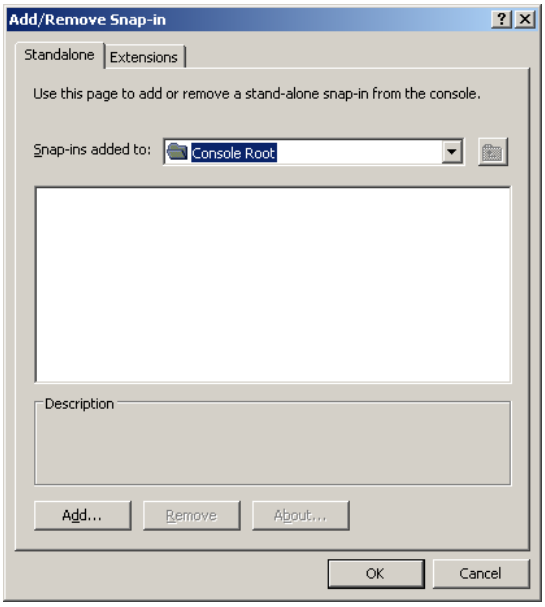
3.3.1 From the Windows task bar, Start > Run > mmc



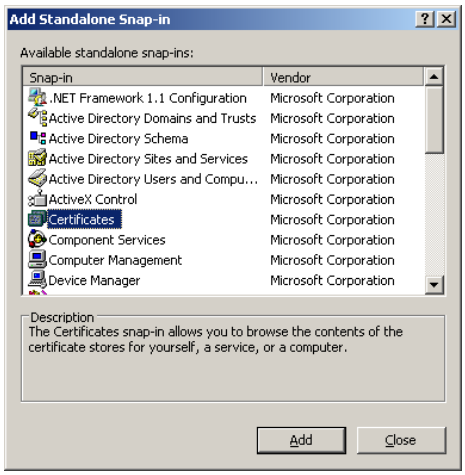
3.3.2 File > Add/Remove Snap-in



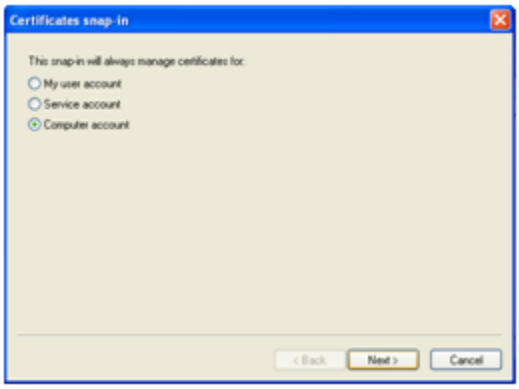
3.3.3 Click “Add…”



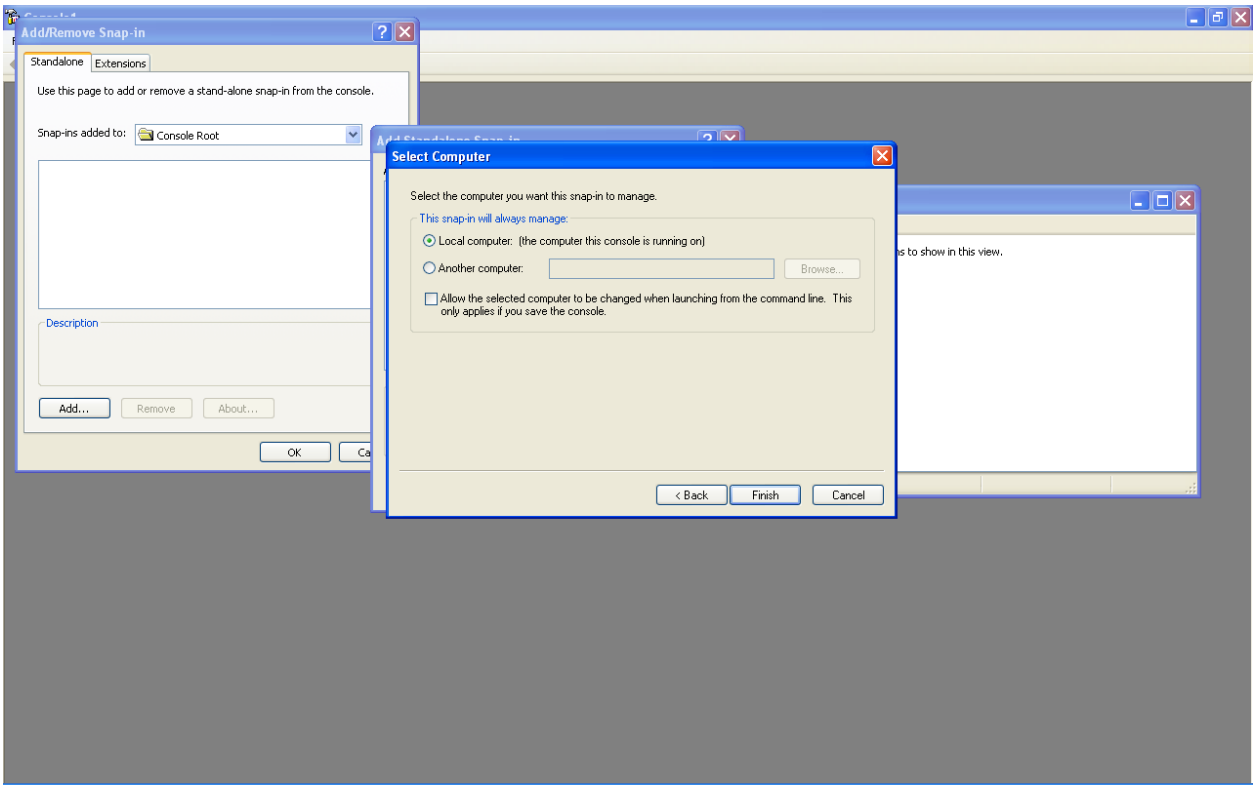
3.3.4 In the “Add Standalone Snap-in”, select “Certificates” and Click “Add”



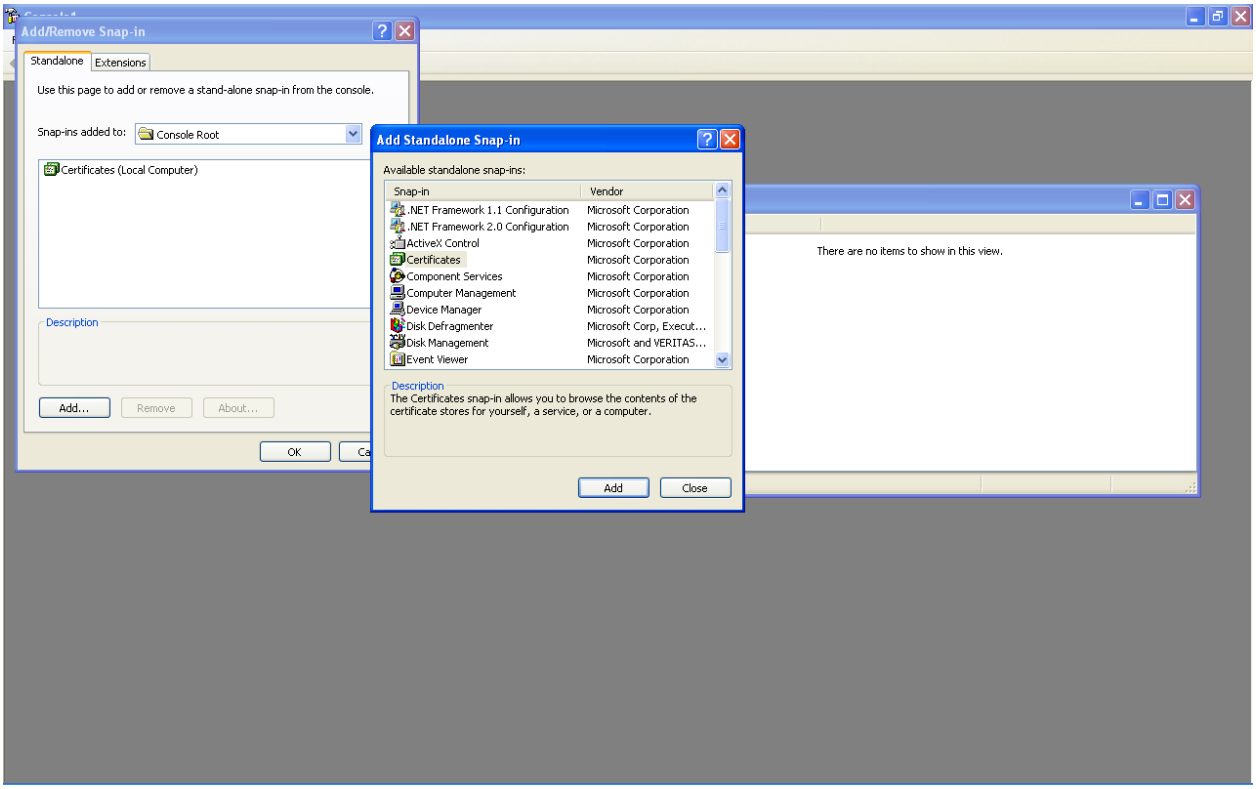
3.3.5 Select “Computer account”, then click “Next”



3.3.6 Select “Local computer (the computer this console is running on)”, and click Finish

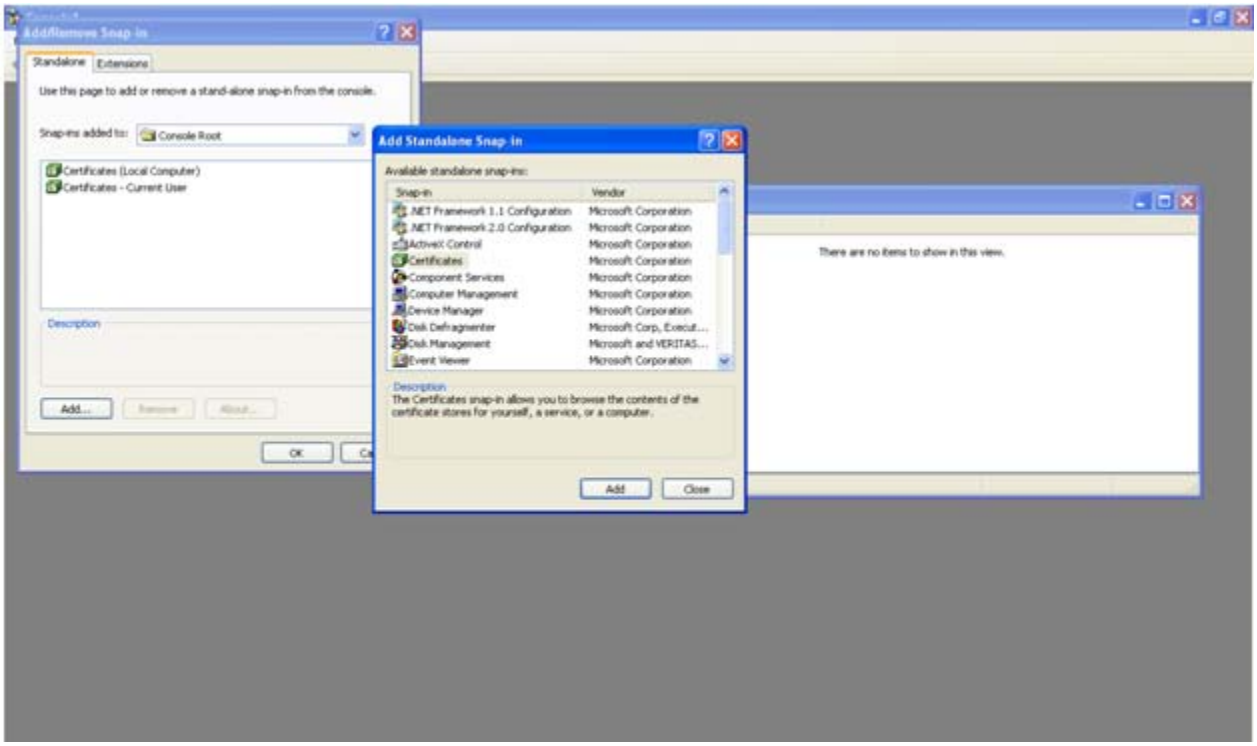

Add/Remove Snap-in
Standalone
Extensions
Use this page to add or remove a stand-alone snap-in from the console.
Snap-ins added to:
Console Root
Description
Add...
Remove
About...
OK
Select Computer
Select the computer you want this snap-in to manage.
This snap-in will always manage:
Local computer: (the computer this console is running on)
Another computer:
Browse...
Allow the selected computer to be changed when launching from the command line. This only applies if you save the console.
< Back
Finish
Cancel
There are no items to show in this view.

3.3.7 Note that the “Certificates (Local Computer)” has been added to the “Add/Remove Snap-in”

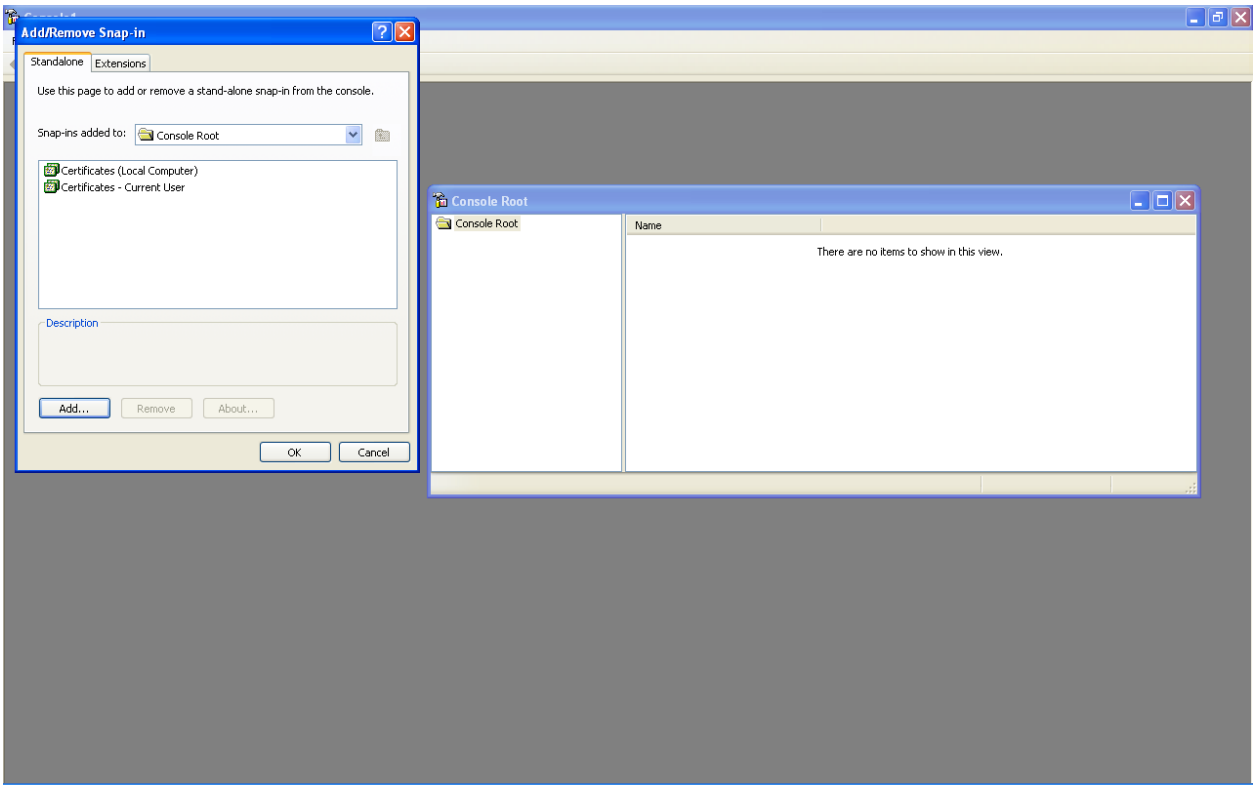


Close the “Add Standalone Snap in”.

3.3.8 In the “Add/Remove Snap-in” click on the “Add..” again and repeat steps 3.3.4 and 3.3.5, but this time for step 3.3.5 select the “My user account”.



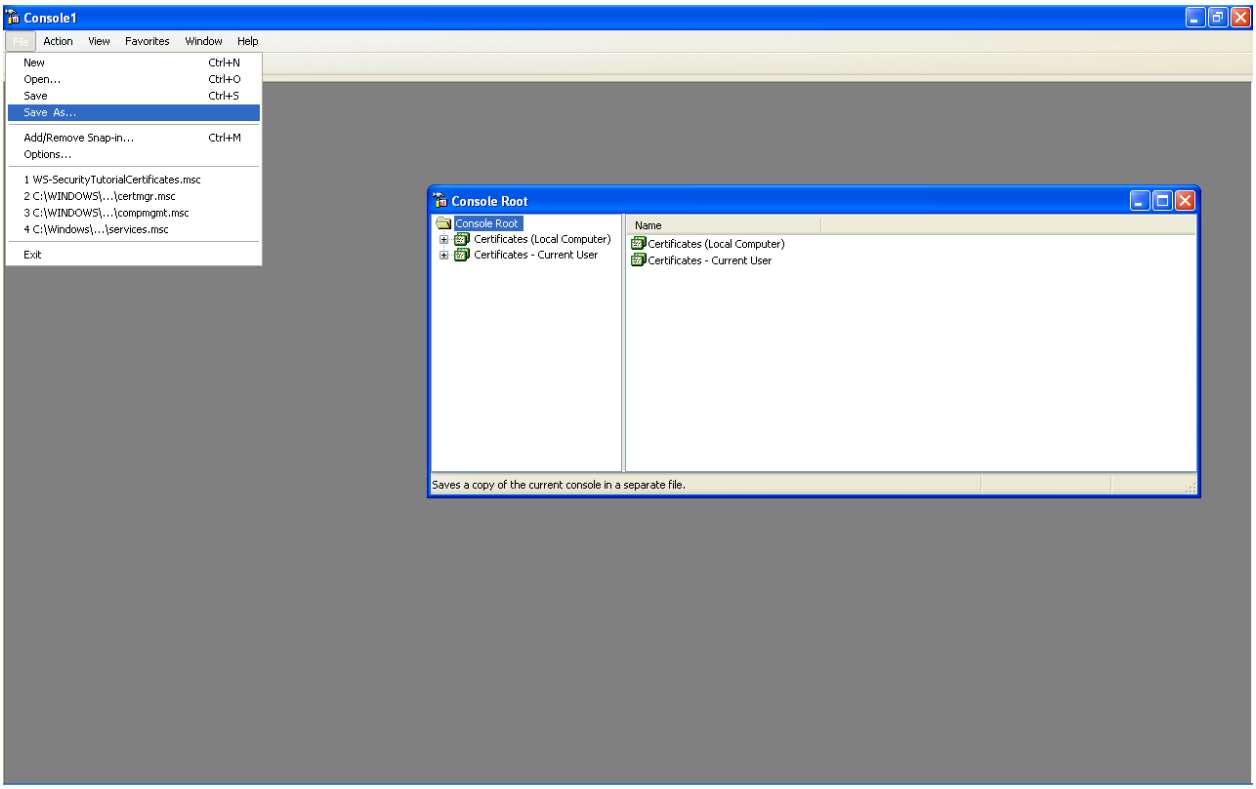
Then Close the “Add Standalone Snap-in”



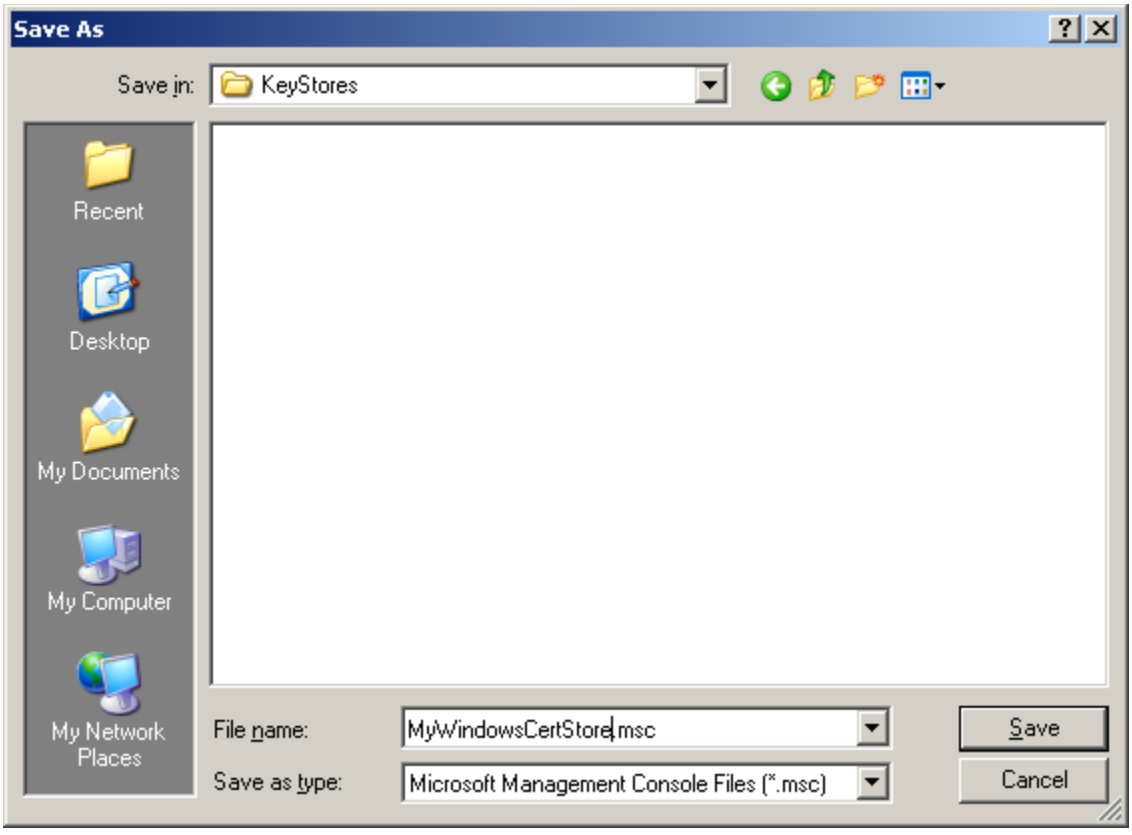
Note that “Certificates – Current User” has also been added to the “Add/Remove Snap-in”

Click on the OK button

3.3.8 Save the mmc console

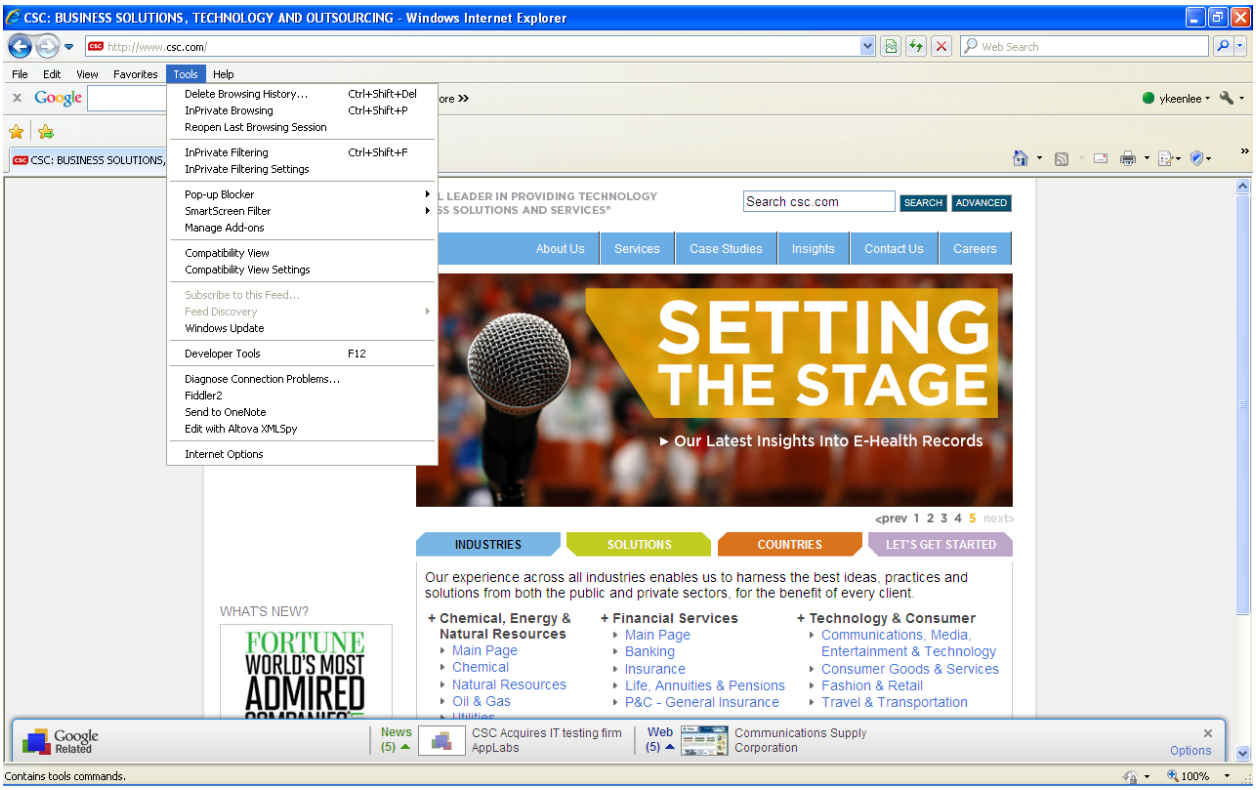


For example, in the **Keystores** folder as "**MyWindowsCertStore**"

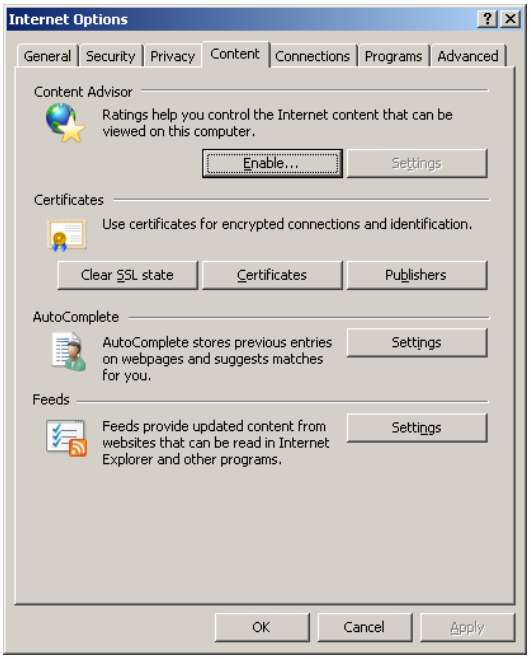


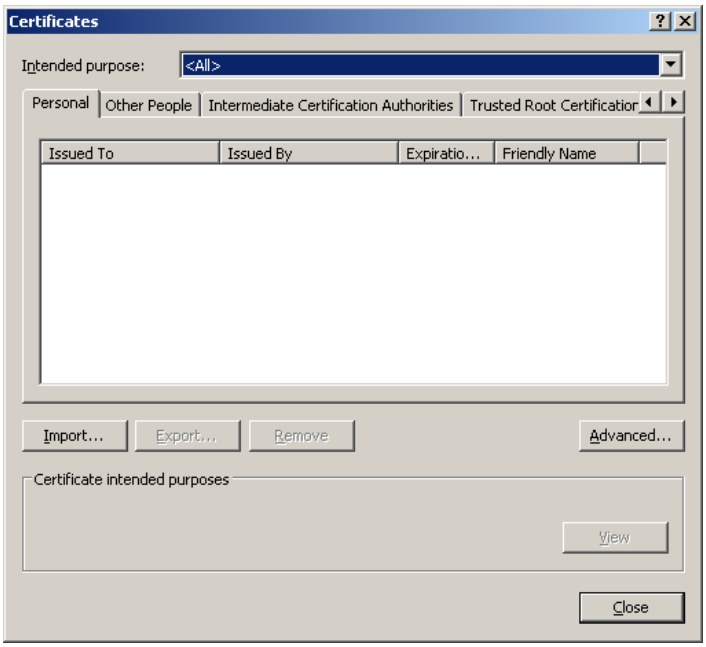
3.4 Importing Certifcates Using the IE Certificates Wizard

Open Internet Explorer go to the Tools > Internet Options menu



Go to the Contents tab and click on the Certificates button



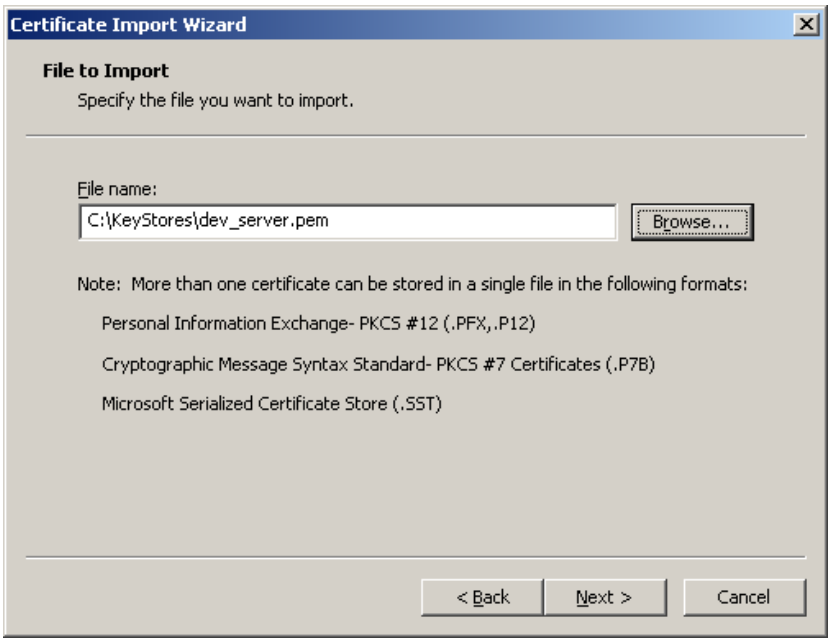


Click on Import…

Certificate Import Wizard
Welcome to the Certificate Import Wizard
This wizard helps you copy certificates, certificate trust lists, and certificate revocation lists from your disk to a certificate store.
A certificate, which is issued by a certification authority, is a confirmation of your identity and contains information used to protect data or to establish secure network connections. A certificate store is the system area where certificates are kept.
To continue, click Next.
< Back
Next >
Cancel

Click the Next button

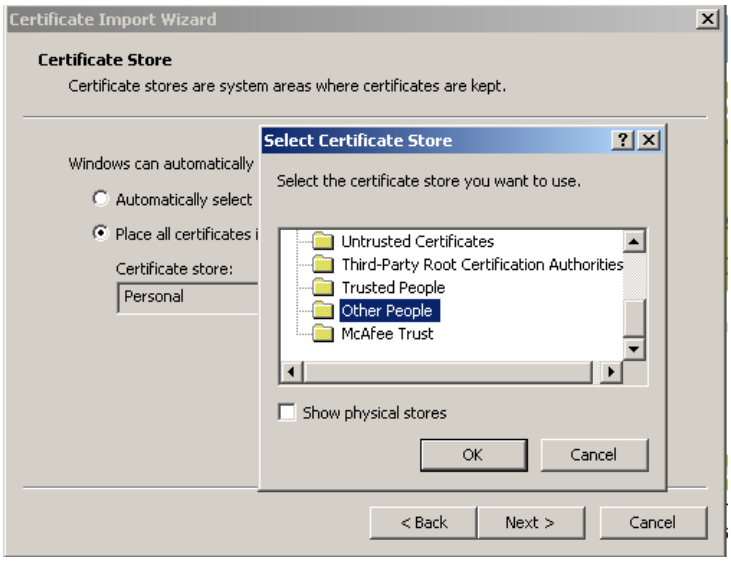
Navigate to the **dev_server.pem** file which you downloaded from eMedNY

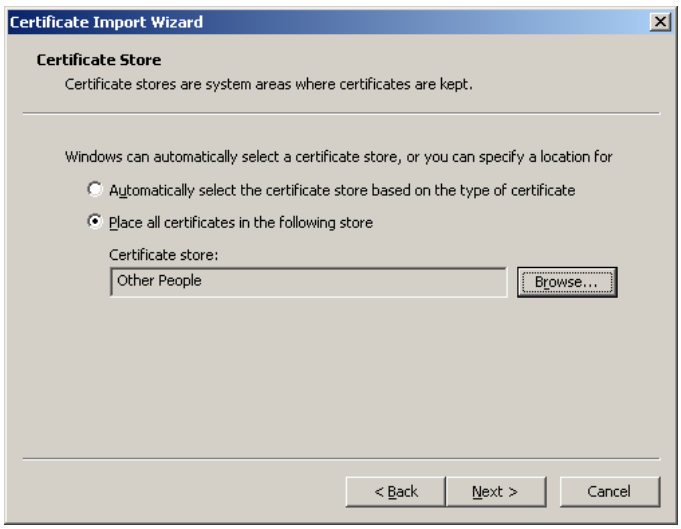

Certificate Import Wizard
File to Import
Specify the file you want to import.
File name:
C:\KeyStores\dev_server.pem
Browse...
Note: More than one certificate can be stored in a single file in the following formats:
Personal Information Exchange- PKCS #12 (.PFX,.P12)
Cryptographic Message Syntax Standard- PKCS #7 Certificates (.P7B)
Microsoft Serialized Certificate Store (.SST)
< Back
Next >
Cancel

Click Next

Notice it says Personal store. Click on Browse, select the “Other People” store. (Note: you can also put it in the "Personal" store, but as it is a server certificate, it might be more appropriate to put it in the "Other People" store and leave the "Personal" store for your own certificates. Also note that these are put into what is known as the "Certificates – Current User" stores. There is also a "Certificates (Local Computer)" stores which can also be used.)

And click OK



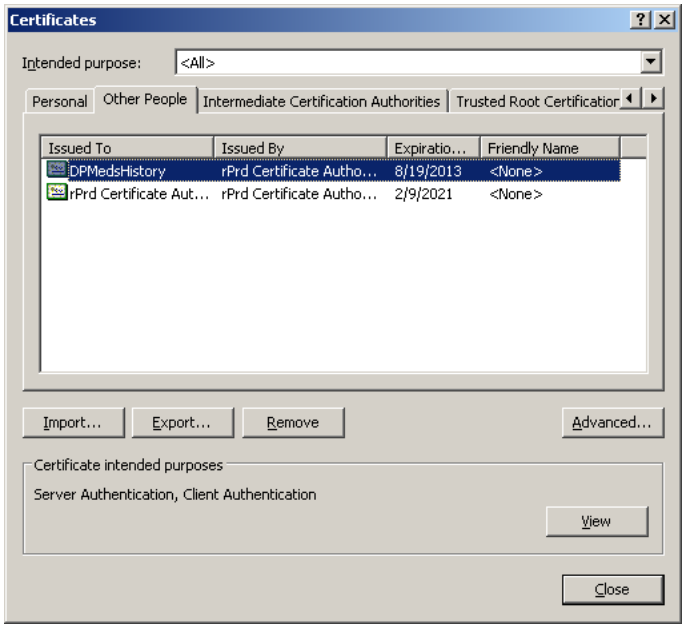

Certificate Import Wizard
Certificate Store
Certificate stores are system areas where certificates are kept.
Windows can automatically select a certificate store, or you can specify a location for
Automatically select the certificate store based on the type of certificate
Place all certificates in the following store
Certificate store:
Other People
Browse...
< Back
Next >
Cancel

Click Next

Certificate Import Wizard
Completing the Certificate Import Wizard
You have successfully completed the Certificate Import wizard.
You have specified the following settings:
Certificate Store Selected by User Other People
Content PKCS #7 Certificates
File Name U:\KeyToolDemo\dev
< Back
Finish
Cancel

Click Finish. Note the import was successful and click OK.

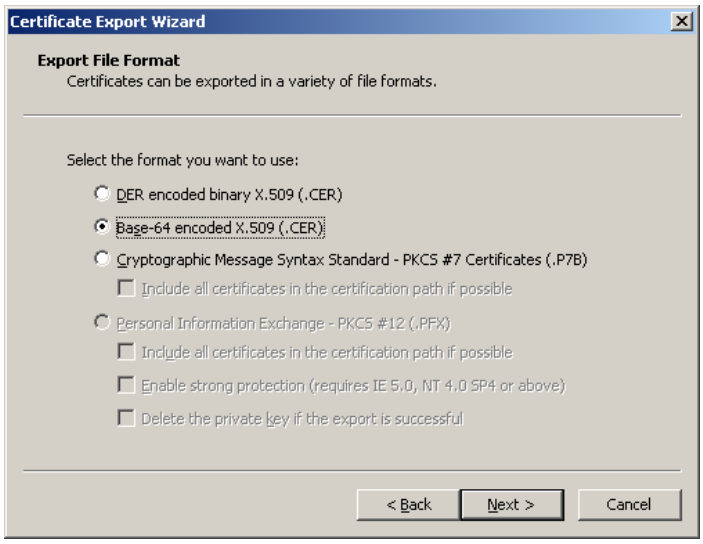
Click on “Other People” tab. The **DPMedsHistory** (which is the name of servercert) issued by “**rPrd Certificate Authority**” (the eMedNY CA) is now in the Windows Certificate Store. Note also that the CA cert in the chain, **rPrd Certificate Authority** is also imported. Click on the **DPMedsHistory** entry to select it. The Export button is now enabled.



Click on the Export… button, and then Next

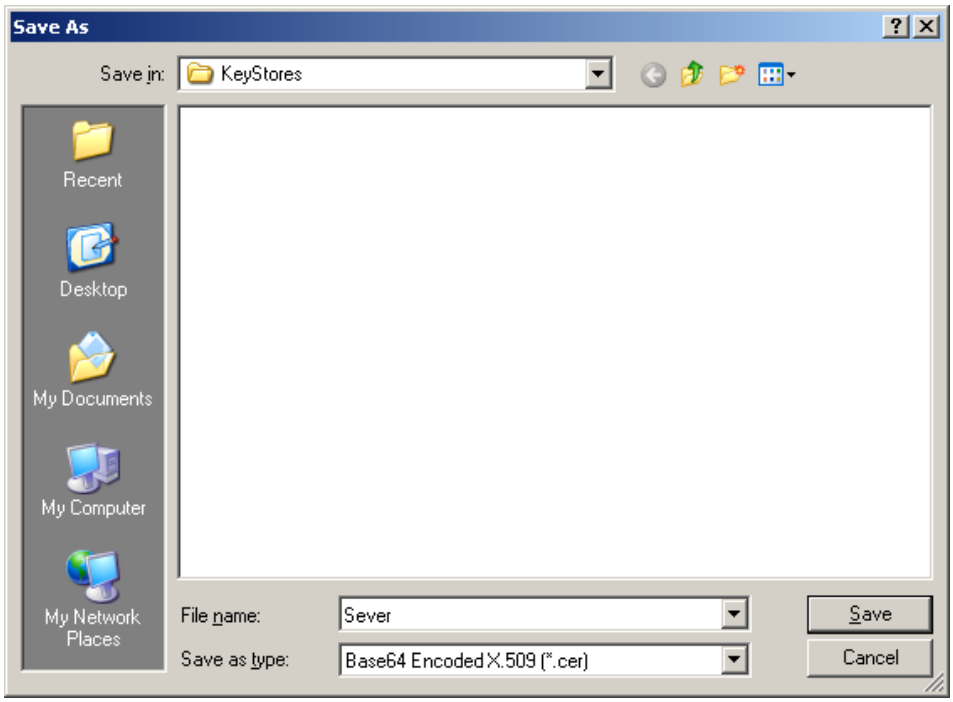
Certificate Export Wizard
Welcome to the Certificate Export Wizard
This wizard helps you copy certificates, certificate trust lists and certificate revocation lists from a certificate store to your disk.
A certificate, which is issued by a certification authority, is a confirmation of your identity and contains information used to protect data or to establish secure network connections. A certificate store is the system area where certificates are kept.
To continue, click Next.
< Back
Next >
Cancel

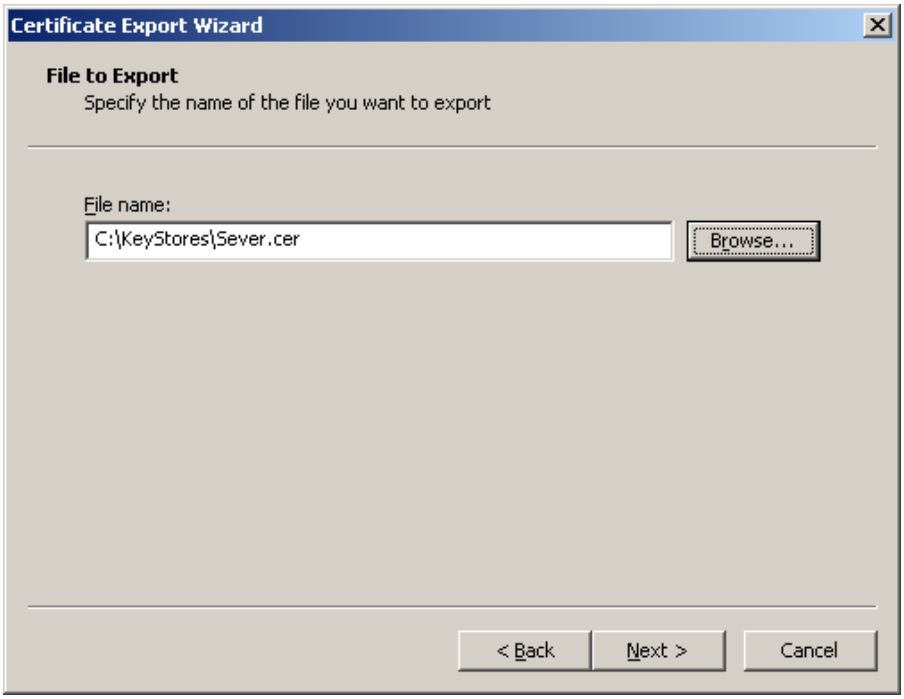
And select the base-64 encoded X.509 radio button,


Certificate Export Wizard
Export File Format
Certificates can be exported in a variety of file formats.
Select the format you want to use:
DER encoded binary X.509 (.CER)
Base-64 encoded X.509 (.CER)
Cryptographic Message Syntax Standard - PKCS #7 Certificates (.P7B)
Include all certificates in the certification path if possible
Personal Information Exchange - PKCS #12 (.PFX)
Include all certificates in the certification path if possible
Enable strong protection (requires IE 5.0, NT 4.0 SP4 or above)
Delete the private key if the export is successful
< Back
Next >
Cancel

Click Next

Navigate to the folder you created for your keystores (in our example “Kestores”) folder and give the exported file a name (for example “**server**” here) with extension “.cer” and Save it.

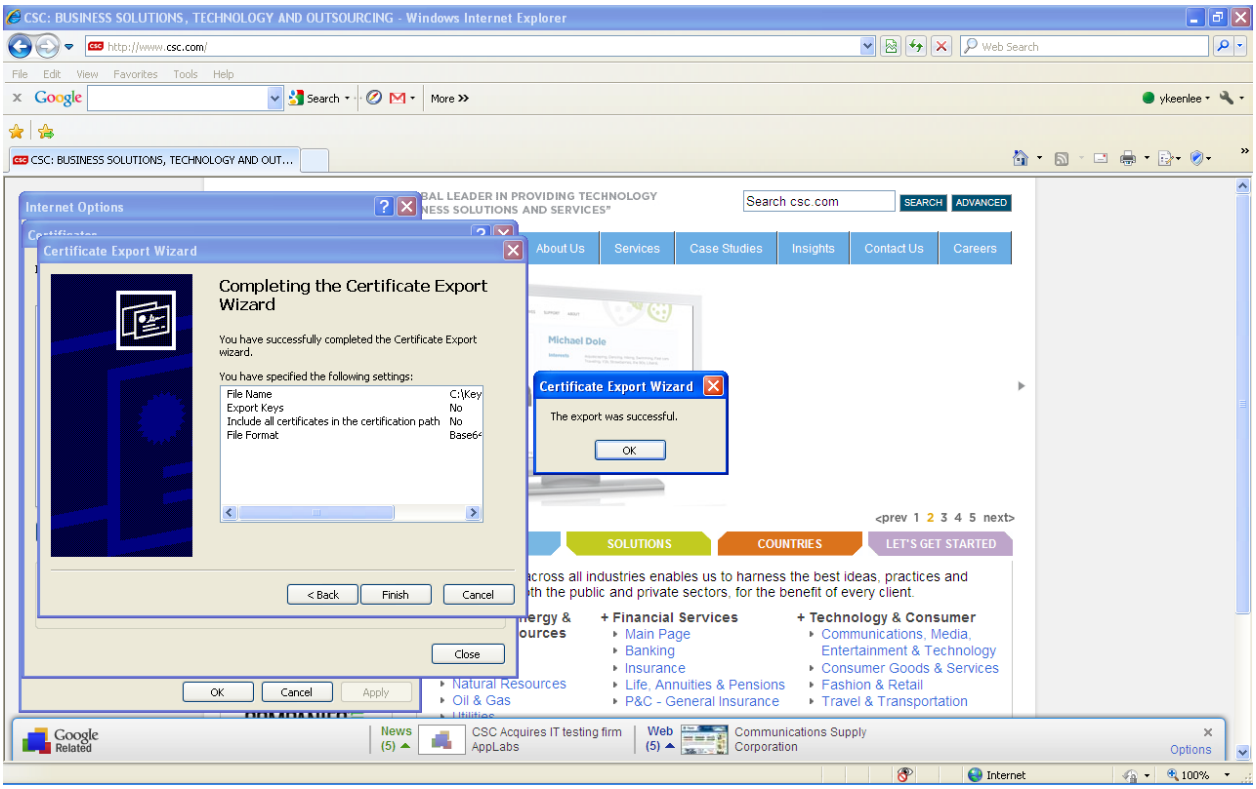




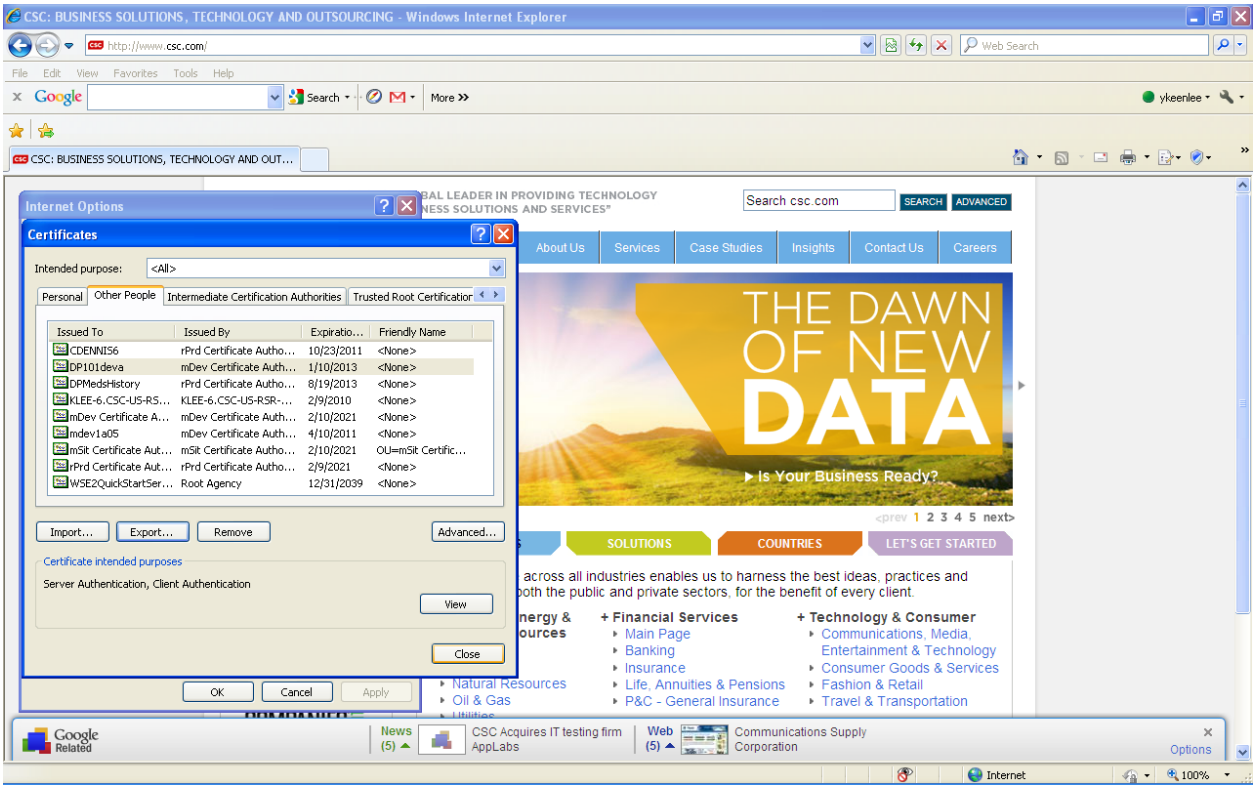
Click Next

Note that the Keys (i.e private keys if any) are not exported and that all certificates in the certifcation path are not included (i.e. the CA cert is not included,) which is what we wanted from this digression using the Windows Certificate Store tool.

Click Finish and note that the export was successful


CSC: BUSINESS SOLUTIONS, TECHNOLOGY AND OUTSOURCING - Windows Internet Explorer
Internet Options
Certificate Export Wizard
Completing the Certificate Export Wizard
You have successfully completed the Certificate Export wizard.
You have specified the following settings:
File Name C:\Key
Export Keys No
Include all certificates in the certification path No
File Format Base64
< Back
Finish
Cancel
Close
OK
Cancel
Apply
Certificate Export Wizard
The export was successful.
OK
Search csc.com
SEARCH
ADVANCED
About Us
Services
Case Studies
Insights
Contact Us
Careers
SOLUTIONS
COUNTRIES
LET'S GET STARTED
+ Financial Services
Main Page
Banking
Insurance
Life, Annuities & Pensions
P&C - General Insurance
+ Technology & Consumer
Communications, Media, Entertainment & Technology
Consumer Goods & Services
Fashion & Retail
Travel & Transportation
Natural Resources
Oil & Gas
CSC Acquires IT testing firm AppLabs
Communications Supply Corporation

Click OK


CSC: BUSINESS SOLUTIONS, TECHNOLOGY AND OUTSOURCING - Windows Internet Explorer
Internet Options
Certificates
Intended purpose: <All>
Personal
Other People
Intermediate Certification Authorities
Trusted Root Certification
Issued To
Issued By
Expiratio...
Friendly Name
Import...
Export...
Remove
Advanced...
Certificate intended purposes
Server Authentication, Client Authentication
View
Close
OK
Cancel
Apply
THE DAWN OF NEW DATA
Is Your Business Ready?
Search csc.com
SEARCH
ADVANCED
About Us
Services
Case Studies
Insights
Contact Us
Careers
SOLUTIONS
COUNTRIES
LET'S GET STARTED
+ Financial Services
Main Page
Banking
Insurance
Life, Annuities & Pensions
P&C - General Insurance
+ Technology & Consumer
Communications, Media, Entertainment & Technology
Consumer Goods & Services
Fashion & Retail
Travel & Transportation
Natural Resources
Oil & Gas
CSC Acquires IT testing firm AppLabs
Communications Supply Corporation

Click Close and then OK.

So now we have the "**server.cer**" exported into the **KeyToolDemo** folder

3.5 Importing the Server Certificate into the Keystore

We can now proceed to import the server certificate into the keystore.

Open the command prompt and navigate to the folder you created for your Keystore. In our example, we named it "Keystores."

Type the following command line:

keytool -importcert -v -alias [Server Cert Alias] -file [Cert file Name].pem -keystore [Keystore name].jks -storepass [keystore password] -keypass [client password]

The following is a sample command:

keytool -importcert -v -alias **Server** -file **Server**.cer -keystore **JDOEKeystores**.jks -storepass **Password1** -keypass **Password2**

This image follows the sample:

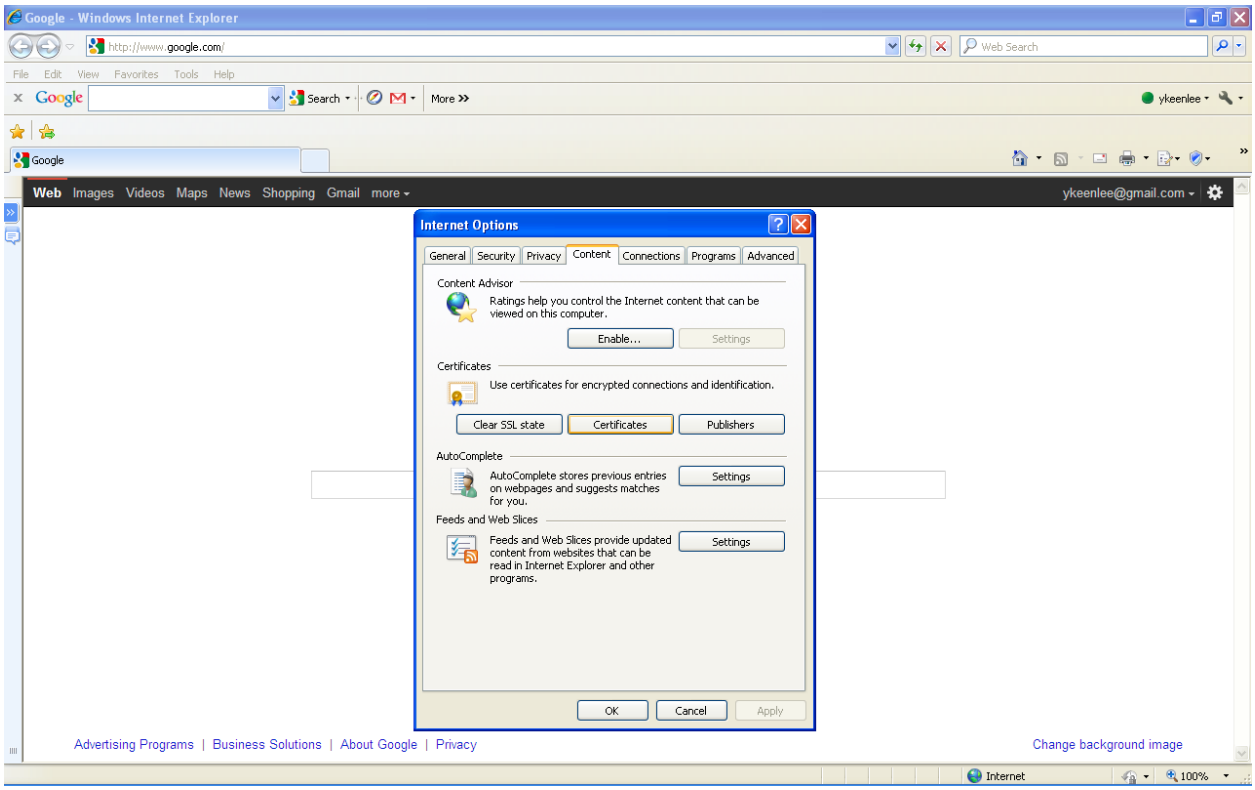
C:\WINDOWS\system32\cmd.exe
C:\KeyStores>keytool -importcert -v -alias JDOE -file JDOE.pem -keystore JDOEKey
stores.jks -storepass Password1 -keypass Password2
keytool error: java.lang.Exception: Certificate reply does not contain public ke
y for <JDOE>
java.lang.Exception: Certificate reply does not contain public key for <JDOE>
at sun.security.tools.KeyTool.validateReply(Unknown Source)
at sun.security.tools.KeyTool.installReply(Unknown Source)
at sun.security.tools.KeyTool.doCommands(Unknown Source)
at sun.security.tools.KeyTool.run(Unknown Source)
at sun.security.tools.KeyTool.main(Unknown Source)
C:\KeyStores>keytool -importcert -v -alias Server -file Server.cer -keystore JDO
EKeystores.jks -storepass Password1 -keypass Password2
Owner: CN=DPMedsHistory, OU=eServers, OU=ePaces, OU=eMedNY-PROD, O=eMedNY
Issuer: OU=rPrd Certificate Authority, O=eMedNY
Serial number: f
Valid from: Tue Feb 16 00:00:00 EST 2010 until: Mon Aug 19 23:59:59 EDT 2013
Certificate fingerprints:
MD5: 05:52:94:C1:75:3C:A1:68:BB:48:D8:BD:08:F1:AF:E2
SHA1: C7:B7:ED:92:61:AB:63:31:D2:AC:71:2F:A8:55:D3:60:F4:A9:BD:6D
Signature algorithm name: SHA1withRSA
Version: 3
Extensions:
#1: ObjectId: 2.5.29.15 Criticality=true
KeyUsage [
DigitalSignature
Non_repudiation
Key_Encipherment
Data_Encipherment
]
#2: ObjectId: 2.5.29.14 Criticality=false
SubjectKeyIdentifier [
KeyIdentifier [
0000: B3 AD 22 1E 93 1C FC 9C F6 17 89 6D FE 51 E7 2D ..".....m.Q.-
0010: 5B 59 72 8D [Yr.
]
]
#3: ObjectId: 2.5.29.31 Criticality=false
CRLDistributionPoints [
[DistributionPoint:
[CN=CRL1, OU=rPrd Certificate Authority, O=eMedNY]
, DistributionPoint:
[URIName: http://www.emedny.org/ePaces/cacerts/CRL1.crl]
]]
#4: ObjectId: 2.5.29.37 Criticality=false
ExtendedKeyUsages [
serverAuth
clientAuth
]
#5: ObjectId: 2.5.29.35 Criticality=false
AuthorityKeyIdentifier [
KeyIdentifier [
0000: C1 BA 37 B5 74 45 72 4D 30 37 98 36 0C F4 BE FF ..7.tErM07.6....
0010: EC 55 1E 74 .U.t
]
]
Trust this certificate? [no]: yes
Certificate was added to keystore
[Storing JDOEKeystores.jks]
C:\KeyStores>

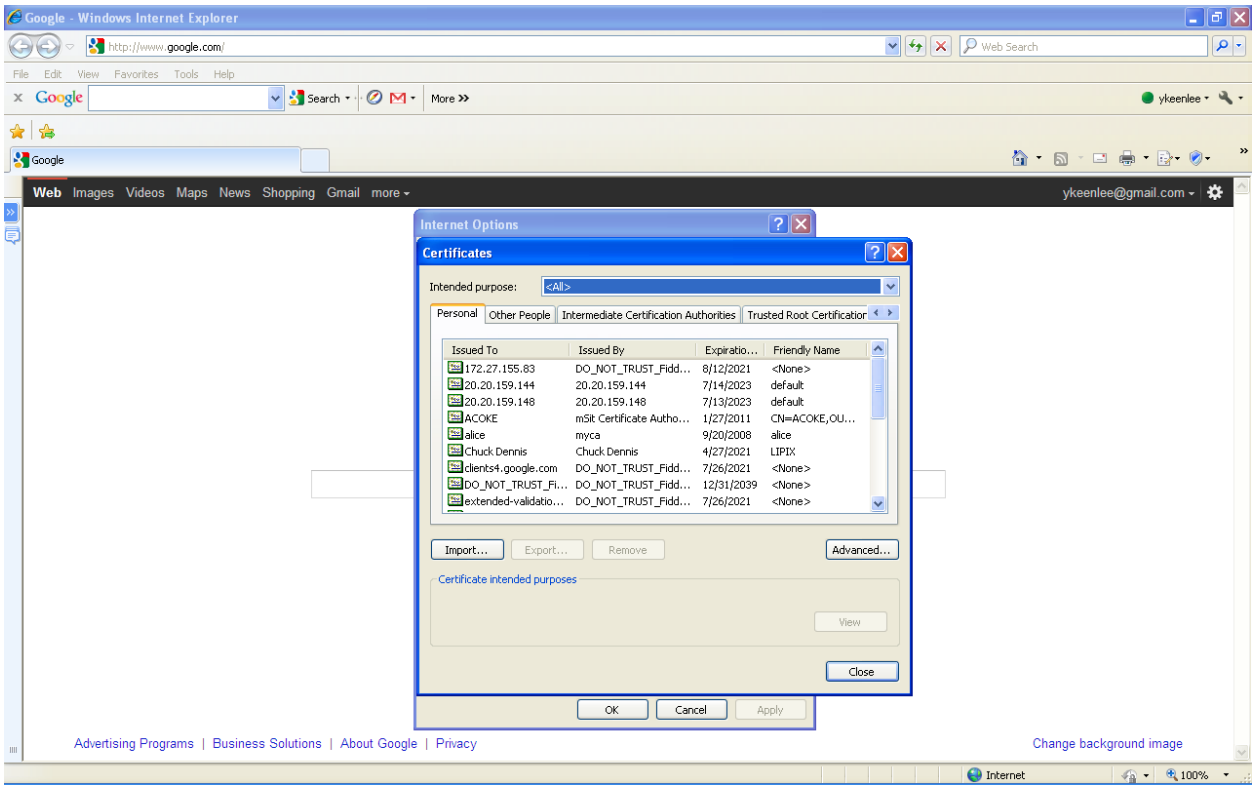
3.6 Importing private key and certificates from Java to Windows Key Stores

If you have a Dotnet application, you will not be able to use the java **JDOEKeystores**.jks so you will have to export the private key and cert of **JDOE** into the Windows Certificate Store. You have already imported the server certificate into that store.

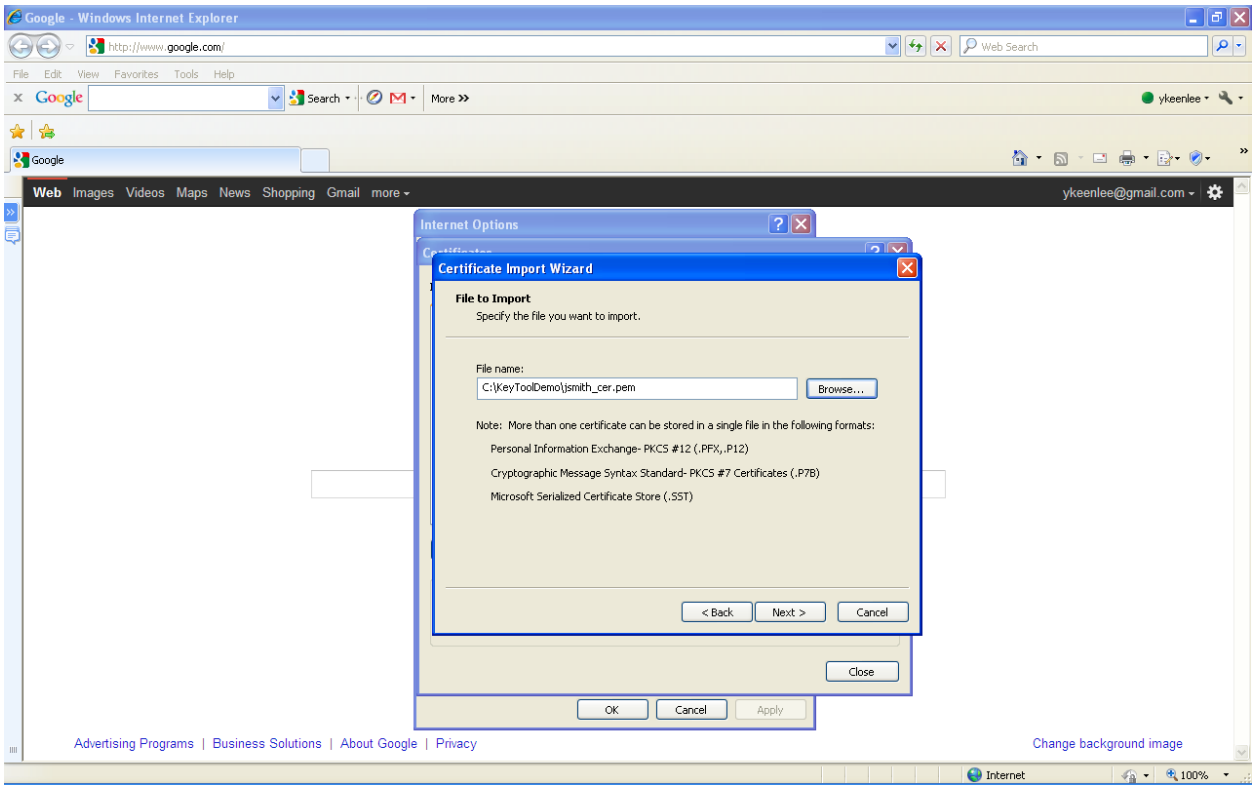
The procedure for doing the **JDOE.pem** is similar.

Open your IE browser, From the menu, go to Tools > Internet options > Content tab and click on Certificates button.

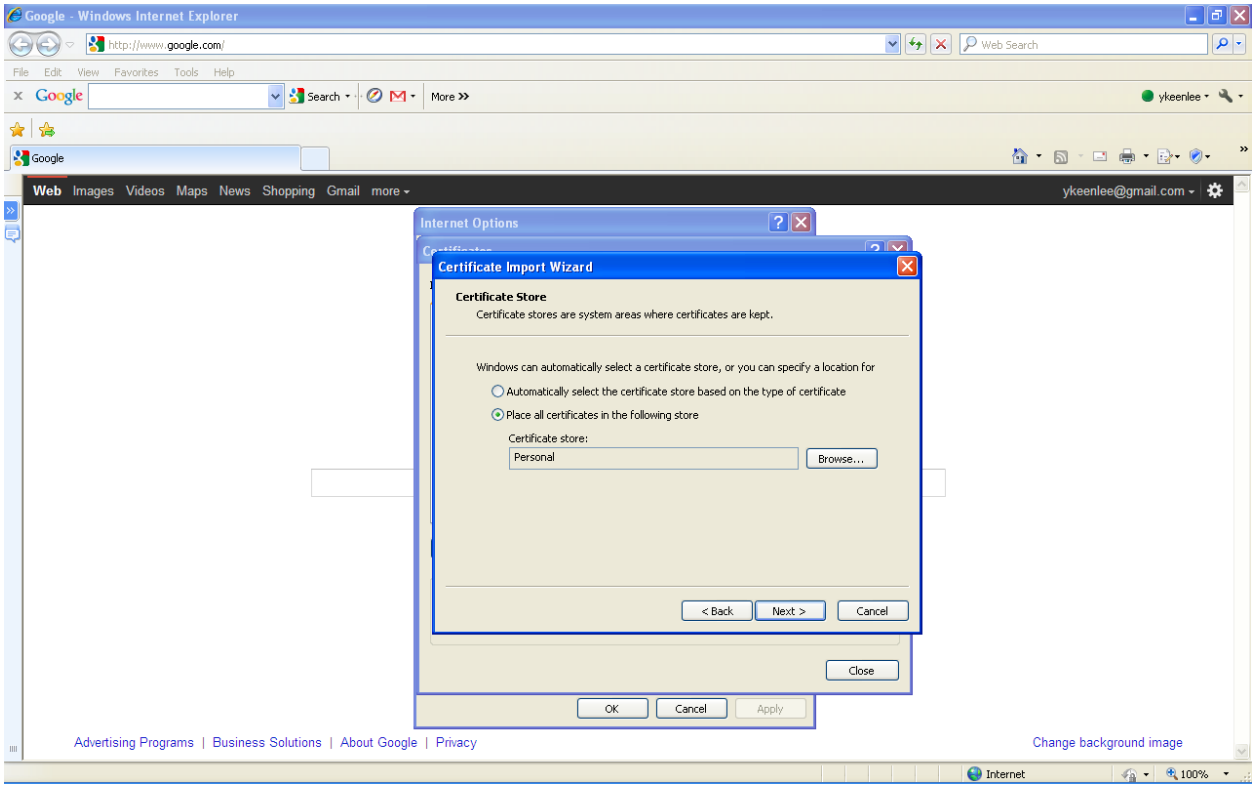




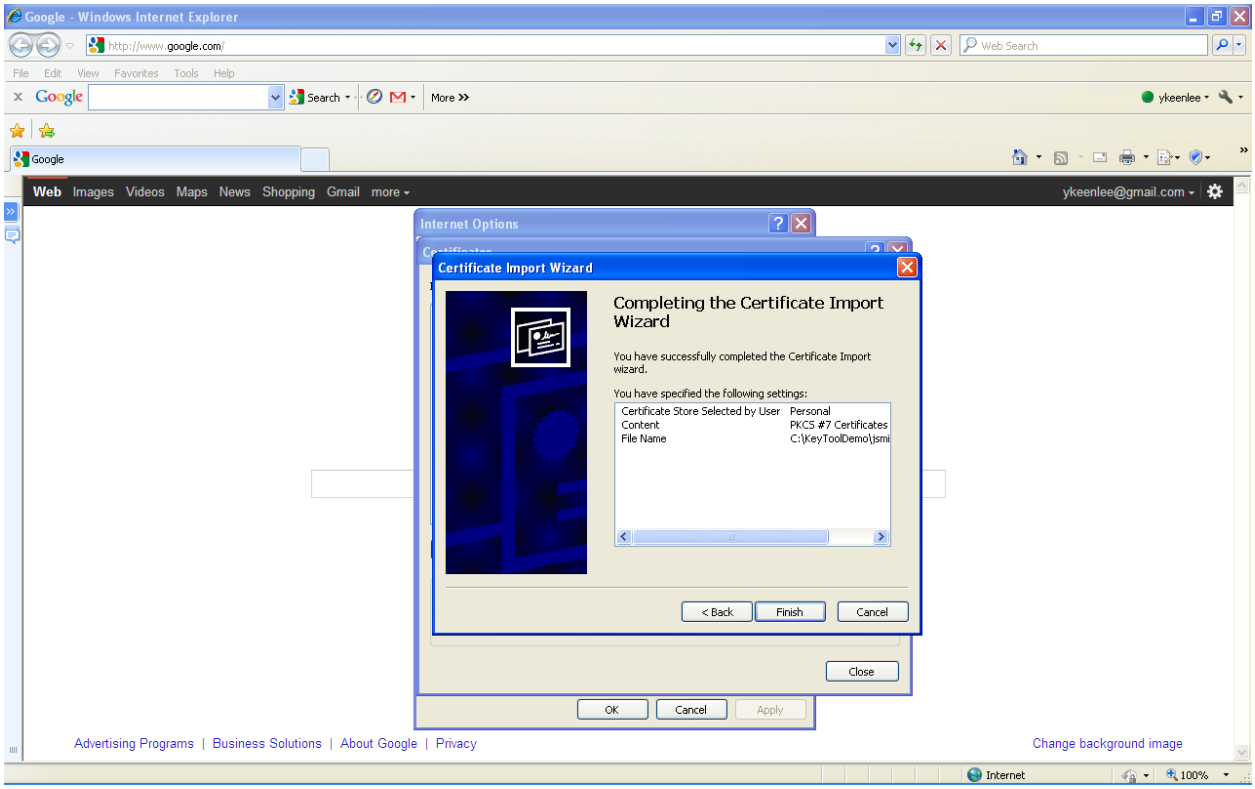
Click Import… , Next and select your **JDOE.pem** file that was returned with your csr request.


Google - Windows Internet Explorer
Certificate Import Wizard
File to Import
Specify the file you want to import.
File name:
C:\KeyToolDemo\jsmith_cer.pem
Browse...
Note: More than one certificate can be stored in a single file in the following formats:
Personal Information Exchange- PKCS #12 (.PFX,.P12)
Cryptographic Message Syntax Standard- PKCS #7 Certificates (.P7B)
Microsoft Serialized Certificate Store (.SST)
< Back
Next >
Cancel
Close
OK
Cancel
Apply

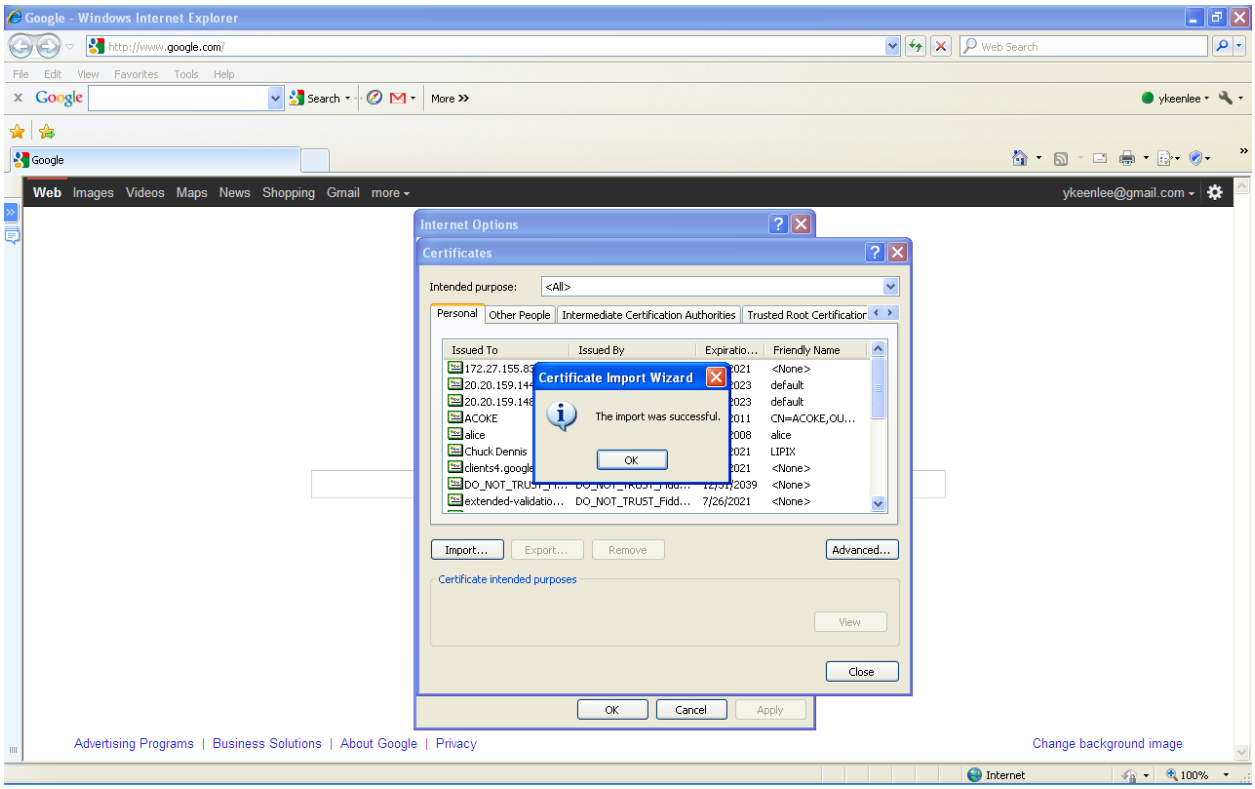
Click Next


Google - Windows Internet Explorer
Certificate Import Wizard
Certificate Store
Certificate stores are system areas where certificates are kept.
Windows can automatically select a certificate store, or you can specify a location for
Automatically select the certificate store based on the type of certificate
Place all certificates in the following store
Certificate store:
Personal
Browse...
< Back
Next >
Cancel
Close
OK
Cancel
Apply

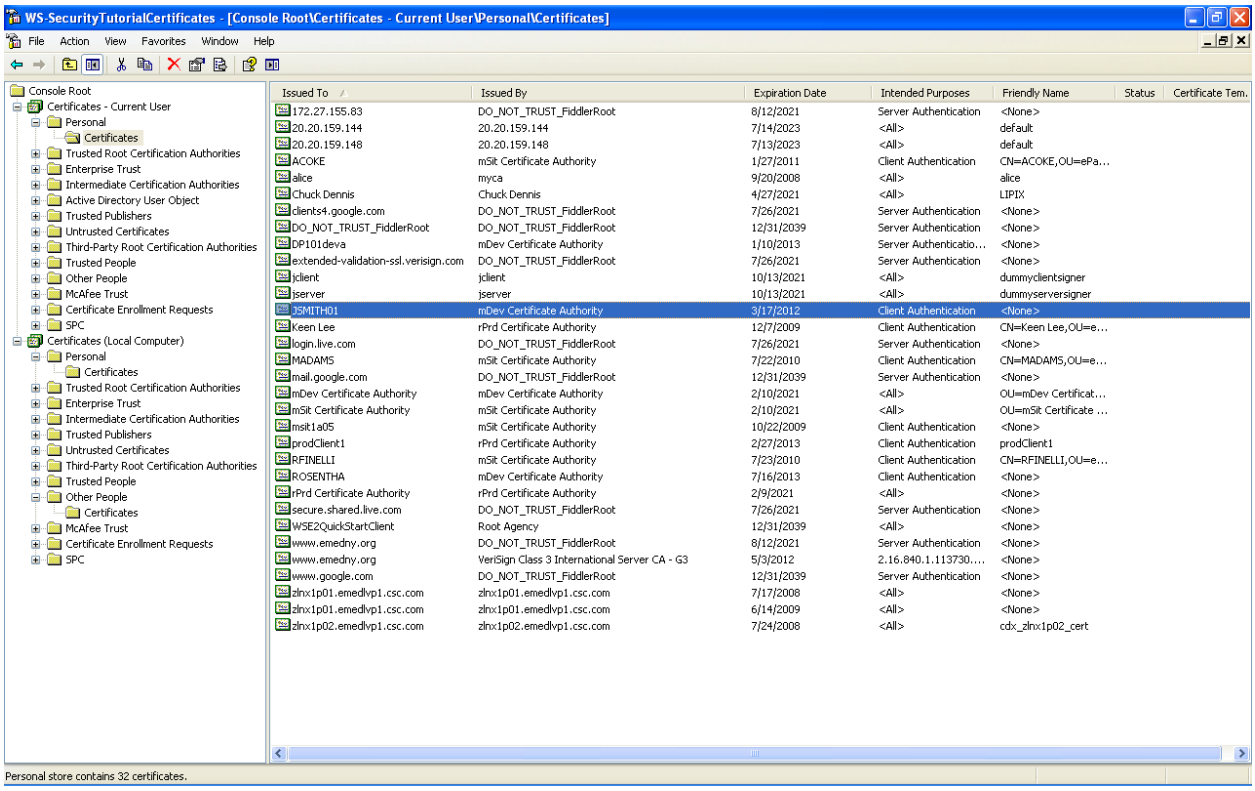
And place the certifcate in the “Personal” store. Click Next



And Finish



You will need to create a mmc certificates to view cert for **JDOE** imported into CurrentUser/Personal/Certificates. The Windows Cert Store only shows CurrentComputer stores.


WS-SecurityTutorialCertificates - [Console Root\Certificates - Current User\Personal\Certificates]
File Action View Favorites Window Help
Console Root
Certificates - Current User
Personal
Certificates
Trusted Root Certification Authorities
Enterprise Trust
Intermediate Certification Authorities
Active Directory User Object
Trusted Publishers
Untrusted Certificates
Third-Party Root Certification Authorities
Trusted People
Other People
McAfee Trust
Certificate Enrollment Requests
SPC
Certificates (Local Computer)
Personal
Certificates
Trusted Root Certification Authorities
Enterprise Trust
Intermediate Certification Authorities
Trusted Publishers
Untrusted Certificates
Third-Party Root Certification Authorities
Trusted People
Other People
Certificates
McAfee Trust
Certificate Enrollment Requests
SPC
Issued To
Issued By
Expiration Date
Intended Purposes
Friendly Name
Status
Certificate Tem.
172.27.155.83 DO_NOT_TRUST_FiddlerRoot 8/12/2021 Server Authentication <None>
20.20.159.144 20.20.159.144 7/14/2023 <All> default
20.20.159.148 20.20.159.148 7/13/2023 <All> default
ACOKE mSit Certificate Authority 1/27/2011 Client Authentication CN=ACOKE,OU=ePa...
alice myca 9/20/2008 <All> alice
Chuck Dennis Chuck Dennis 4/27/2021 <All> LIPIX
clients4.google.com DO_NOT_TRUST_FiddlerRoot 7/26/2021 Server Authentication <None>
DO_NOT_TRUST_FiddlerRoot DO_NOT_TRUST_FiddlerRoot 12/31/2039 Server Authentication <None>
DP101deva mDev Certificate Authority 1/10/2013 Server Authenticatio... <None>
extended-validation-ssl.verisign.com DO_NOT_TRUST_FiddlerRoot 7/26/2021 Server Authentication <None>
jclient jclient 10/13/2021 <All> dummyclientsigner
jserver jserver 10/13/2021 <All> dummyserversigner
JSMITH01 mDev Certificate Authority 3/17/2012 Client Authentication <None>
Keen Lee rPrd Certificate Authority 12/7/2009 Client Authentication CN=Keen Lee,OU=e...
login.live.com DO_NOT_TRUST_FiddlerRoot 7/26/2021 Server Authentication <None>
MADAMS mSit Certificate Authority 7/22/2010 Client Authentication CN=MADAMS,OU=e...
mail.google.com DO_NOT_TRUST_FiddlerRoot 12/31/2039 Server Authentication <None>
mDev Certificate Authority mDev Certificate Authority 2/10/2021 <All> OU=mDev Certificat...
mSit Certificate Authority mSit Certificate Authority 2/10/2021 <All> OU=mSit Certificate ...
msit1a05 mSit Certificate Authority 10/22/2009 Client Authentication <None>
prodClient1 rPrd Certificate Authority 2/27/2013 Client Authentication prodClient1
RFINELLI mSit Certificate Authority 7/23/2010 Client Authentication CN=RFINELLI,OU=e...
ROSENTHA mDev Certificate Authority 7/16/2013 Client Authentication <None>
rPrd Certificate Authority rPrd Certificate Authority 2/9/2021 <All> <None>
secure.shared.live.com DO_NOT_TRUST_FiddlerRoot 7/26/2021 Server Authentication <None>
WSE2QuickStartClient Root Agency 12/31/2039 <All> <None>
www.emedny.org DO_NOT_TRUST_FiddlerRoot 8/12/2021 Server Authentication <None>
www.emedny.org VeriSign Class 3 International Server CA - G3 5/3/2012 2.16.840.1.113730.... <None>
www.google.com DO_NOT_TRUST_FiddlerRoot 12/31/2039 Server Authentication <None>
zlnx1p01.emedlvp1.csc.com zlnx1p01.emedlvp1.csc.com 7/17/2008 <All> <None>
zlnx1p01.emedlvp1.csc.com zlnx1p01.emedlvp1.csc.com 6/14/2009 <All> <None>
zlnx1p02.emedlvp1.csc.com zlnx1p02.emedlvp1.csc.com 7/24/2008 <All> cdx_zlnx1p02_cert
Personal store contains 32 certificates.

4 Additional Tools and Information

4.1 keytool web link

http://java.sun.com/j2se/1.3/docs/tooldocs/win32/keytool.html

4.2 Requirements for CORE Compliance

http://www.caqh.org/benefits.php

4.3 JSSE Reference Guide

http://docs.oracle.com/javase/6/docs/technotes/guides/security/jsse/JSSERefGuide.html#SSLOverview

4.4 WCF – 2 Way SSL using Certificates

http://blogs.msdn.com/b/imayak/archive/2008/09/12/wcf-2-way-ssl-security-using-certificates.aspx

eMedNY neither endorses nor recommends any of the tools linked or referenced in this document. The intent here is strictly informational.

eMedNY is the name of the electronic New York State Medicaid system. The eMedNY system allows New York Medicaid providers to submit claims and receive payments for Medicaid-covered services provided to eligible clients.

eMedNY offers several innovative technical and architectural features, facilitating the adjudication and payment of claims and providing extensive support and convenience for its users. CSC is the eMedNY contractor and is responsible for its operation.

The information contained within this document was created in concert by eMedNY DOH and eMedNY CSC. More information about eMedNY can be found at www.emedny.org.